**DOCTOR OF SCIENCES**

# Microsimulation in time and space
# applications and challenges

Dumont, Morgane

*Award date:*
2021

*Awarding institution:*
University of Namur


**UNIVERSITÉ DE NAMUR**

FACULTÉ DES SCIENCES

DÉPARTEMENT DE MATHÉMATIQUE

# Microsimulation in time and space: applications and challenges

Thèse présentée par
**Morgane Dumont**
pour l'obtention du grade
de Docteur en Sciences

Composition du Jury:

Arnaud BANOS
Timoteo CARLETTI (Promoteur)
Éric CORNÉLIS (Co-Promoteur)
Philippe LIÉGEOIS
Catherine LINARD
Germain VAN BEVER (Président du Jury)

Mai 2021

Graphisme de couverture : ©Presses universitaires de Namur

Rue Grandgagnage 19
B-5000 Namur (Belgique)



Imprimé en Belgique

ISBN : 978-2-39029-143-5
Dépôt légal: D/2020/1881/12

Université de Namur
Faculté des Sciences
rue de Bruxelles, 61, B-5000 Namur (Belgique)

**Microsimulation temporelle et spatiale: applications et challenges**
par Morgane Dumont

**Résumé :** La microsimulation permet la conception et l'évolution temporelle et spatiale de populations d'individus statistiquement similaires à une population cible. Cette thèse en propose deux applications concrètes : la localisation de débris spatiaux qui pourraient endommager des satellites en service, ainsi que l'estimation des besoins en soins de santé des personnes âgées habitant en Belgique (jusqu'en 2030). Ces applications ont généré des questions méthodologiques développées et traitées ensuite. En effet, nous montrons que l'ordre choisi pour appliquer les sous-modèles dans l'évolution d'une population, à temps discret, d'un an, influence le résultat obtenu. Nous diminuons alors la variabilité des résultats en proposant une alternative consistant à fixer des dates pour les événements pertinents. Ces études d'évolution de populations synthétiques ont ouvert la voie à des questions théoriques, d'application à des contextes plus généraux, comme la classification de données déséquilibrées et en présence de données non observées. Ces dernières analyses ont été réalisées en développant et en comparant des réseaux de neurones et un modèle de choix discret "logit".

**Microsimulation in time and space: applications and challenges**
by Morgane Dumont

**Abstract:** Microsimulation aims at mimicking a real population under scrutiny and simulating its temporal and spatial evolution. This thesis proposes two applications related to actual problems: the creation of a synthetic population of space debris, causing potential damages to functional satellites and the forecast of the health needs of the elderlies for Belgium until 2030. Then, the observed limitations of the methods generate new methodological research questions. Hence, the impact of specific orders of the application of the sub-models for a discrete time simulation with a fixed timestep of a synthetic population is investiguated and quantified. Then, we propose the calendar based approach consisting in fixing birthdays for each agent and a date of death for the dying agents. This reduces the variability of the results. These studies about the evolution of synthetic populations induced theoretical questions whose scope goes beyond the presented framework. Indeed, noticing that simulating the occurrence or not of an event is equivalent to perform a binary classification, we delved into the problem of highly unbalanced classes with unobserved variables. These achievements have been obtained by developing and comparing a feedforward neural network and a logit discrete choice model.

Thèse de doctorat en Sciences Mathématiques (Ph.D. thesis in Mathematics)
Date: 17/05/2021
Département de Mathématique
Promoteurs (Advisors): Timoteo CARLETTI, Eric CORNÉLIS

# Remerciements

Bien que d'apparence personnelle, cette thèse est en réalité le résultat de beaucoup de super rencontres, en étant entourée par des personnes bienveillantes et aidantes. Ce document comprend le résultat final, mais ce qui compte n'est pas tant la destination, mais plutôt le chemin parcouru pendant ces nombreuses années, conférences, collaborations, échanges, ... Je vais donc essayer, dans cette section, de remercier chaque personne qui a participé, de près ou de loin, à l'accomplissement de ce chemin. J'espère n'oublier personne...

En tout premier lieu, je souhaite remercier mes promoteurs, qui ont tous deux eu un rôle fondamental pour moi dans cette thèse. Éric, sans qui ce contrat de la région Wallonne (que je remercie également) n'aurait pas vu le jour, ainsi que Téo, que je ne connaissais encore pratiquement pas, mais qui a accepté de me prendre en thèse en même temps que ce contrat. C'est donc un peu par "hasard" qu'ils sont devenus mes promoteurs, mais quelle chance j'ai eue ! Vous avez toujours été disponibles lorsque j'en avais besoin, mais sans me mettre aucune pression. Une fois le contrat de la région Wallonne terminé, j'ai été libre de prendre les directions que je souhaitais, de faire les collaborations qui m'intéressaient, en vous sachant toujours prêts à m'aider et à me soutenir. Au-delà d'une bienveillance professionnelle, vous avez été compréhensifs sur ma situation personnelle, me permettant de lever le pied quand cela s'avérait nécessaire. En bref, merci, merci et encore merci, vous m'avez aidée à grandir en tant que chercheuse tout en me permettant un bon équilibre entre la vie professionnelle et privée.

J'aimerais remercier mon jury, qui m'a donné des conseils précieux pour m'aider à améliorer cette thèse, mais également pour la suite des recherches. Je suis ravie que vous ayez accepté de faire partie de mon jury, chacun dans votre domaine d'expertise et que la discussion ait pu être aussi constructive. Merci à mon comité d'accompagnement et aux collaborateurs du contrat de la RW, tels que Thierry, Mélanie, Jean-Paul, Philippe Toint, Véronique Tellier, Dominique Dubourg,... Merci!

"Yo man !", ça y est, I dit it ! :-) Même si cela fait longtemps, je me souviens très bien Jojo, t'avoir promis lors de mon premier séjour en Australie d'écrire "Yo man" dans ma thèse. Tu auras été un super soutien également, dès le début, alors qu'on se connaissait à peine. Tu étais dans le rush de présenter ta thèse et partir à l'autre bout du monde, mais tu as quand même pris du temps pour moi, pour m'expliquer Virtual-Belgium, pour me donner des conseils, m'assurer que je pourrai t'écrire si j'avais des problèmes d'installations. Et même si cela pouvait te paraitre insignifiant, ça m'a beaucoup rassurée de savoir que tu étais là si nécessaire. Tu es également une super rencontre (et June aussi !). Merci de m'avoir acceuillie dans ton bureau, dans votre foyer, de m'avoir fait découvrir tant de choses. Merci à June pour son accueil également lors de mes séjours. Vous êtes au top tous les deux ! Merci !

En parlant de super acceuil en Australie, je souhaite également remercier toute l'équipe SMART à Wollongong et remercier chaleureusement Pascal Perez et Tania Brown, ainsi que Shiva Pedram.

Un merci particulier à toi, Arnaud Banos, qui aura été présent aux moments clés de ma thèse. Tout d'abord expert pour la région Wallonne, ensuite rencontré en Australie et en conférence, et enfin jury de ma thèse. Chaque rencontre aura été constructive. Merci pour tes idées et tes conseils.

Merci à Anne Lemaître, Alexis P. et Daniel de m'avoir permis de découvrir ce domaine tout nouveau pour moi, ainsi que l'importance des simulations de débris spatiaux. Merci de m'avoir donné la possibilité de faire cette recherche interdisciplinaire très intéressante.

Merci à ma famille pour le soutien, tout particulièrement à Simon pour avoir géré presque tout à la maison me permettant d'utiliser les forces que la grossesse me laissait pour la thèse. Eloïse pour son sourire et sa bonne humeur m'obligeant à faire des pauses régulières. Merci à mon frère, Michaël, et Ruta d'avoir pris certains jours de congé pendant qu'Eloïse était malade pour venir la garder et me permettre d'avancer. Merci à ma famille complète mais également à la famille de Simon. Merci à tous nos amis également, qu'ils soient ou non des facs ! Merci à la team Arsenal pour tous ces midis partagés.

Je souhaite également remercier André Hardy et Germain Van Bever. Être votre assistante est un plaisir ! Merci à Martine de prendre du temps pour la répartition des cours et d'être bienveillante dans cette gestion, y compris au niveau des repos de maternité.

Merci à Martin G. et Julien B. pour les nombreuses collaborations pour les cours, les nombreuses discussions et leur soutien. Merci à la Team du "Bureau d'en face", par laquelle sont passés Jérémy, Jon, Martin, Watson et Alexis. Venir parler deux minutes avec vous faisait souvent plaisir et était toujours très amusant. Un merci particulier à Martin pour nos échanges de cours nous permettant nos séjours à l'étranger, pour

sa gentillesse, son humour et surtout, pour me rappeler le syndrôme de l'imposteur lorsque celui-ci m'atteignait.

Merci à William et Dorothée d'avoir été présents tant sur le plan professionnel que personnel ! Merci à Candy, Marie et William pour ces années de bureau partagées dans la bonne humeur (et avec du chocolat xD).

Merci à Alice, Pascale, Juan et Frédéric Wautelet pour le soutien administratif/ technique, mais aussi pour leur disponibilité, leur sourire et leur bonne humeur. On est toujours bien reçu quand on a besoin de vous.

Merci à Manon, Ambi, Nicolas (en particulier pour ton humour et tes jeux de mots que tu m'expliques souvent gentiment), Arnaud, Marie Moriamé, Julien P., Delphine, Pauline, Eve (qui veille à notre santé mentale XD), Mara, Loïc, Joanna,..., d'être de super collègues !

Je remercie l'université de Namur et plus particulièrement le département de Mathématique pour l'acceuil et l'atmosphère familiale.

Merci au Dr Mercier, qui est un médecin de famille exceptionnel s'intéressant réellement à ses patients, en considérant les projets qu'ils ont, tels que, par exemple, présenter une thèse à un moment particulier.

Last but not least, I would like to thanks Robin Lovelace for this wonderful collaboration on the book. You directly trusted me for helping you with this huge challenge, and I directly trusted you for working on the book. Also thanks for inviting me in Leeds (thanks to you and your colleagues, boss,...) and then for the courses we gave together (I have a specific good memory of Seville).

Et tout simplement, merci à chaque personne qui m'a permise d'en arriver là. J'espère n'avoir oublié personne.

# Contents

# Introduction

## Context

Simulating the temporal and spatial evolution of complex systems plays a major role in a very large range of domains : health (Banos et al., 2015), economics (Bourguignon and Spadaro, 2006), chemistry (Gillespie, 1977), celestial mechanics (Le Maistre et al., 2018), demography (Sabourin and Bélanger, 2015), traffic (Barthélemy and Carletti, 2017*b*), etc. It allows forecasting the effect of different scenarios and thus to adapt the strategies to reach a target, such as for example, fight against a pandemic or perform realistic pension policies. The benefit of validated models is widely recognized, but designing and calibrating such a framework can result in a complex task. Indeed, it needs to be adapted to the data availability and the research goals, while allowing computational simulations in a reasonable time. Moreover, depending on the availability of data and of the confidentiality of diverse information, creating a synthetic populations, statistically similar to the real one (in terms of the aggregated data in use) is sometimes required.

Navigating in the vast possibilities of complex systems modelling includes deterministic and stochastic methods; micro and macroscopic scales; aggregated and totally disaggregated studies; continuous and discrete time; etc. Furthermore, complex systems could involve a combination of different types of dynamics, requiring to mix different types of methods on the same simulation.

A large range of microsimulations and dedicated softwares, mostly analysing populations of humans, already exist. We can for example cite the LIAM (life-cycle income analysis model) framework (O'Donoghue et al., 2009) that contains different types of modules at a microlevel and is calibrated to ensure the fitting of the simulation (re aggregated) to macrodata. This framework is used for instance for the development of the microsimulation "MIDAS_BE" (Dekkers et al., 2009) considering simultaneously 3 types of modules: the demographic one, the labour market and the pension scheme for three different countries: Belgium, Germany and Italy. An updated

and faster version of this framework is LIAM2 (de Menten et al., 2014) used for the discrete-time dynamic microsimulation model for Luxembourg "MIDAS_LU" (Liégeois, 2021).

This thesis initially presents applications and then methodological questions induced by the applications. Our first result is to present the outcome of a microsimulation scheme applied to determine the evolution of a family of space debris, usually modelled exclusively with deterministic differential equations. Our approach is able to overcome some of the limitations the latter models do exhibit. Then, it also relies on the developed demographic dynamic model which differs from the others already present in the literature by the presence of the grouping into households and municipalities and by the total control we keep on the complete process since it is entirely coded within our platform. Finally, this thesis initiates some methodological questions that are discussed in the last chapters.

# Structure of the thesis

The thesis is articulated in three distinct parts.

### Part I : Preliminaries

First, some preliminaries are developed (Chapter 1) to fix the global context and set the thesis in the literature. We describe in this part some theoretical concepts such as microscopic and macroscopic models, continuous time and discrete time simulations, deterministic and stochastic modelling. Finally, the well-known method of Iterative Proportional Fitting is briefly exposed.

### Part II : Two main applications

Then, we contributed in two concrete applications during this research, explained in the second part.

The first application (Chapter 2) deals with the construction of a population of spatial debris in the Geostationary Earth Orbit. This chapter illustrates the adaptation capacity of the synthetic populations methods. This part of the thesis allowed us to analyse a totally unknown field never studied before.

Chapter 3 consists in creating a tool aiming at simulating the Belgian population from 2011 until 2030 and was initially developed in a project with the Walloon Region (VBIH). Thanks to the delivered framework, the aim of the project is to foresee better social and health conditions for elder people in the next decades.

**Part III : Methods**

Finally, the third part questions some common practices in population simulations, and takes a step back to analyse the impact of some choices done in the previous simulation part. Chapter 4 considers a validated framework, called TransMob (Huynh et al., 2016), presenting a population evolution for a region of Australia similar to the one we did for VBIH and highlights the impact of changing the order of the dynamical procedures (ageing, passing away, giving birth, divorcing, getting married) needed to make the initial population evolve. Then, an alternative approach is proposed and called "the calendar based" approach.

A second point capturing our attention is the design of models to forecast binary choices. Indeed, it is very common to forecast if the individual in the microsimulation will perform or not a given action (for example, getting married, divorcing, have a baby, etc.). To the best of our knowledge, no clear analysis advising specific method for specific configurations exists. Chapter 5 contains a first attempt to test two methods (Discrete Choice modelling and Back-propagated Artificial Neural Networks) in several specific configurations such as very unbalanced classes, or missing information.

The manuscript is concluded with a discussion of the results obtained with this research and some possible perspectives for future works.

# Programming languages

The work performed in this thesis requires a lot of implementations coded in different programming languages, depending on the research fields, the specific needs and the collaborators. Chapter 2 contains simulations in Fortran, being the standard software used in Celestial Mechanics, while the analysis of the results have been done using Python. The initial synthetic population presented in Chapter 3 is performed in R, whereas the dynamical evolution needing object oriented compiled coding is in C++. The analysis of the order of the procedures in Chapter 4 is performed in R; however, because the calendar based approach has been added to the existing platform (Huynh et al., 2016), coded in Javascript by their creators, we also used the latter. Python is chosen for the study of Chapter 5. Finally, the first simulations of continuous time in the final discussion are performed in Matlab. Note that, for the divorces' modelling (in Chapter 3 and in Chapter 5), Biogeme (Bierlaire, 2003) is employed.

# Contributions

Contributions directly associated with the thesis are reported in this section. Several proceedings and articles (all peer-reviewed) have been published:

- M. Dumont, J. Barthelemy, N. Huynh, T. Carletti (2018) Towards the Right Ordering of the Sequence of Models for the Evolution of a Population Using Agent-Based Simulation. *Journal of Artificial Societies and Social Simulation, 21(4)*

- M. Dumont, J. Barthelemy, T. Carletti, N. Huynh (2017), Importance of the order of the modules in TransMob [Huynh et al., 2015], *Proceedings - 22nd International Congress on Modelling and Simulation*, p 811-817

- M. Dumont, J. Barthelemy, T. Carletti (2017), Robustness of artificial neural network and discrete choice modelling in presence of unobserved variables, *Proceedings - 22nd International Congress on Modelling and Simulation*, p 480-486)

- M. Dumont, T. Carletti, E. Cornélis (2017), Population synthétique: un outil pour une analyse spatiale fine des besoins futurs en soins de santé, In S. Carbonnelle, T. Eggerickx, V. Flohimont, S. Perelman, & A. Vandenhooft (Eds.), *Vieillissement et entraide: Quelles méthodes pour décrire et mesurer les enjeux ?* (Vol. 6, pp. 55-74). Presses Universitaires de Namur (PUN).

Moreover, spatial microsimulation being interdisciplinary and useful in many domains, a book regrouping the important notions of this field has been written :

- R. Lovelace, M. Dumont (2016), Spatial Microsimulation with R, *CRC Press. 260 p. (Chapman & Hall/CRC The R Series)*

This book includes the codes to create synthetic populations in R and is accessible for non mathematicians.

Finally, a collaboration for the creation of a synthetic population of space debris results in these peer-reviewed articles :

- A. Petit, D. Casanova, M. Dumont, A. Lemaitre (2018), Creation of a synthetic population of space debris to reduce discrepancies between simulation and observations, *Celestial Mechanics and Dynamical Astronomy, Springer*, Vol 130, N 12, p. 79

- A. Petit, D. Casanova, M. Dumont, A. Lemaitre (2017), Design of a synthetic population of geostationary space debris by statistical means, *Spaceflight Mechanics*, Vol 160, p 3451-3462

# Acknowledgment

Computational resources have been provided by the Consortium des Équipements de Calcul Intensif (CÉCI), funded by the Fonds de la Recherche Scientifique de Belgique (F.R.S.-FNRS) under Grant No. 2.5020.11 and by the Walloon Region. Moreover, Chapter 3 is part of a project with the support and funding of the Public Service of Wallonia (DGO6), under Grant No. 1318077.

# Part I

# Preliminaries

Chapter 1

# Theoretical concepts

To the best of our knowledge, there exists no clear and exhaustive introduction to the field of synthetic populations and spatial microsimulation, providing a clear way to code the methods in an accessible programming language. Consequently, at the early stage of the thesis, we wrote a book in collaboration with Dr Robin Lovelace [(1)], containing the basis of the field and all the codes needed to perform simple microsimulation project with other input data (Lovelace and Dumont, 2016). To allow everyone to easy access the book, a free version is available online here :`https://spatial-microsim-book.robinlovelace.net/index.html`. The general theoretical concepts relevant to understand the work presented in this thesis are defined briefly in this chapter, based on the book. When a specific chapter needs a new theoretical concept, the latter is explained at the moment we use it.

## 1.1 Macrosimulation, agent-based modelling and microsimulation

Macrosimulation considers a system, or a set of systems, as a whole and the evolution of the population is defined by equations dealing with global quantities on how the full system will evolve (based for example on rates and differential equations). The analysed quantity can be, for instance, an indicator such as the total number of individuals in a specific category of the population. On the other side, microsimulation and agent-based modelling focus on each individual behaviour, considering independently each unit. Note that the terms "population" and "individual" refers to the statistical definition of the concept, meaning that it is not restricted to evolution of population of humans, but it could also be animals, chemical cells or objects. Histor-

[(1)] Associate Professor of Transport Data Science, University of Leeds, `https://www.robinlovelace.net/`

ically, microsimulation regroups a sample and marginals to deduce the population at the individual level. The progress in microsimulation allows to generate this population without sample.

The differences and similitude of micro and macrosimulation are discussed in (Van Imhoff and Post, 1998). The basic microsimulation implies static methods to generate the individuals at one point of time and space, whereas macrosimulation focus on the evolution of the global population, with only the initial indicators needed as starting point. Depending on the research goal, it is important to consider the two possible options and balance the benefits of both methods. In some cases, the simulation involves several dynamical processes interacting with each other and some of those need a microsimulation, whereas other are better modelled by a macrosimulation. This is the case in several hybrid models such as for example (Banos et al., 2015) or (El Hmam et al., 2006).

Agent-based modelling differs formally from microsimulation, since the individuals (here called agents) are interacting with each other and with their environment. This implies time and space moves. However, the differences behind the terminology "Agent-based modelling" and "microsimulation" are quite fuzzy. Some research groups consider as microsimulation everything focused on individuals, meaning that agent based modelling would be a part of microsimulation. In addition, both concepts are often used together or one after the other. In (Bae et al., 2016), they consider both fields with evolving time and differentiate the domains by the fact that microsimulation uses rates on each individuals (whereas macrosimulation applies rates on the count of individuals) and agent-based modelling specifically examines the interaction between the agents, with rates intervening when two agents meet. In (Brown and Harding, 2002), microsimulation encompasses agent-based modelling, and thus consider interactions. A more precise analysis of the terminology "microsimulation" is also available in (Lay-Yee and Cotterell, 2015).

We hope that with the above discussion it is now clear that our work doesn't need the specific label of microsimulation or agent-based modelling, but can be considered as both depending on the referred definition. However, this thesis is definitely not inserted in the field of macrosimulation.

## 1.2 Deterministic and stochastic modelling

When implementing and analysing a model or a process, it is fundamental to be aware of the differences between deterministic and stochastic modelling. Indeed, deterministic simulations are fully determined and same input conditions always lead to same outcomes.

On the other side, stochastic modelling includes some randomness (coming for example from the fact that an event has a given probability, for instance 0.5, to occur

and therefore will happen in (asymptotically) one simulation out of 2). In this case, exactly the same inputs could generate small (or big) differences in the output. When performing a stochastic simulation, it will be important to consider several replicas and analyse the sensitivity of the model to the randomness implicit in the process.

## 1.3 Discrete and continuous variables

When choosing a model, it is important to notice if the variable(s) we try to model is(are) either discrete or continuous. Indeed, simulating, for example, a quantity of liquid or a number of individuals is different, because the quantity of liquid is a continuous variable and the number of individuals is a discrete variable (only integers). This characteristic influences the available choices of models since when analysing a discrete variable (for example the actual number of individuals in a system) depending on time, non differentiable functions appear, which implies to forget a wide range of mathematical methods. To face this problem, one can decide to first smooth the process and simulate a model with non integer number of individuals (often the case in macrosimulation modelled thanks to differential equations, for example when simulation an epidemic model such as (Franco, 2020) or (Kumar et al., 2020)).

## 1.4 Discrete and continuous time simulations

Another important feature of simulations is the way to consider the time (Ossimitz and Mrotzek, 2008). Indeed, in reality, time is a continuous variable, but implementing continuously the time could be a challenge. Even when solving numerically deterministic differential equations, a timestep could be unavoidable, depending on the chosen way to solve the problem.

A possible option is to consider the time as discrete and consider the static state of the population at specific times. This raises a new input parameter : the timestep, which could be constant or evolving. To calibrate this new parameter, the time-scale of the other input data (such as for example the rate associated with the probability to divorce) needs to be considered (Rogers et al., 2014). Moreover, with fixed timestep, the desired horizon of the forecast intervenes. When simulating a population over 10 years, a timestep of 1 minute requires high computational performances. Note that an oversized timestep results in possible missed events (since usually each event could occur only once over a timestep), whereas an undersized timestep involves many steps without any events and thus a simulation which is unnecessarily computationally expensive.

An in-between alternative is possible and already developed in fields such as chemical reactions (Gillespie, 1977) or in demography to determine couples (Zinn, 2012). This method separates the events of the process in two steps : first, choose the moment of the next event and then choose which event occurs at this specific moment.

The choice of which time definition to use is crucial. This thesis first considers population evolving with a yearly timestep, all rates being annual. However, when generating a discrete time simulation, one needs to also determine the order of the sub-models, i.e. the different actions an agent can perform (to age, to die, to give birth, etc) and this choice of the order has an impact on the simulation output (see Chapter 4). The possibility and the associated difficulties to turn into a continuous time evolution based on the Gillespie method are discussed at the end of the thesis.

## 1.5 Iterative Proportional Fitting (IPF)

Having discussed primary concepts about the model, this section will now briefly explain the Iterative Proportional Fitting method that is used in different parts of the thesis. The Iterative Proportional Fitting (IPF) is an intuitive and well-established method first proposed by (Deming and Stephan, 1940) to achieve the goal of building an initial population. Initially, IPF aims at updating the contingency table computed from a given a sample, in a way to respect known marginals, called "the constraints". Historically, the method was used with a global sample of individuals included in the whole population and specific constraints for each small zone of the territory. The process of IPF is thus repeated for each spatial zone. The method is useful since trying to solve these kinds of problem "by hand" would generate a combinatorial problem with a lot of different possibilities. Several adapted versions appeared later, such as for example, for studies without sample available (Barthélemy and Toint, 2013) and (Lenormand and Deffuant, 2013). Moreover, exactly the same idea can be adapted to multidimensional cross-tables as developed, for example in (Pritchard and Miller, 2009). This section briefly introduces the basic IPF. To implement the algorithm in multiple dimensions and also without a sample, an R package is available (Barthélemy and Suesse, 2015).

Two types of data are necessary for the basic IPF: a sample on which all variables are recorded and, the number of individuals per modality of each variable (the marginal distributions). To make the explanation easy to understand, a small example is developed here. Table 1.1 illustrates an example of a sample composed by 11 individuals belonging to two groups Male/Female and to two age classes with one row per person. This table is statistically equivalent to the contingency table of both available variables, with only the ID of the individual lost (see Table 1.2).

Moreover, constraints are often the marginals or the counts for some crossed variables, that could be for example :

- Number of Males : 25
- Number of Females : 30
- Number of 0-50 : 21
- Number of 50-100 : 34

| ID | Gender | Age class |
|---|---|---|
| 1 | Male | 0-50 |
| 2 | Male | 0-50 |
| 3 | Female | 50-100 |
| 4 | Male | 0-50 |
| 5 | Male | 0-50 |
| 6 | Male | 50-100 |
| 7 | Female | 50-100 |
| 8 | Male | 0-50 |
| 9 | Female | 0-50 |
| 10 | Female | 50-100 |
| 11 | Female | 0-50 |

Table 1.1 – Sample example - Individual version

| | Males | Females | Total |
|---|---|---|---|
| 0-50 | 5 | 2 | 7 |
| 50-100 | 1 | 3 | 4 |
| Total | 6 | 5 | 11 |

Table 1.2 – Sample example - contingency table

The ultimate goal of the spatial microsimulation here is to find a population that is statistically similar to the actual population, thus respecting the constraints and with a high probability to observe the used sample when performing a global survey. The sample aims in slightly determining the correlations between the variables. The method could be interpreted as the replication of the individuals in the sample to reach a new larger population that fits the constraints.

IPF will repeat the step explained here until a stopping criteria (either a maximum number of iterations or the fit of the constraints). To illustrate the starting point of the iterative process, Table 1.3 involves the starting cross-table and the constraints.

The IPF considers successively each variable and weight the cross-table to exactly fit the associated constraint.

If we begin with the gender, Table 1.4 calculates each new weight as a proportion of the actual weight. This proportion is simply the ratio between the target and current totals of the variable in consideration (here gender). After this sub-step of the iteration, the gender constraints are exactly respected. However, the target marginals of the age variables are not.

| | Males | Females | Current total | Target total |
|---|---|---|---|---|
| 0-50 | 5 | 2 | 7 | 21 |
| 50-100 | 1 | 3 | 4 | 34 |
| Current total | 6 | 5 | 11 | |
| Target total | 25 | 30 | | 55 |

Table 1.3 – Example - all entry data

| | Males | Females | Current total | Target total |
|---|---|---|---|---|
| 0-50 | $\frac{5*25}{6} = 20.8$ | $\frac{2*30}{5} = 12$ | 32.8 | 21 |
| 50-100 | $\frac{1*25}{6} = 4.2$ | $\frac{3*30}{5} = 18$ | 22.2 | 34 |
| Current total | 25 | 30 | 55 | |
| Target total | 25 | 30 | | 55 |

Table 1.4 – Example - Iteration 1, variable 1 : the gender

The second step is similar but considers the age variable, to constrain then the column marginals (see Table 1.5). This step damages the fit of the gender constraint, whilst resulting in a perfect fit of the age variable. Having here only two variables, this ends the first iteration. The process is then repeated until a stopping criteria is met.

| | Males | Females | Current total | Target total |
|---|---|---|---|---|
| 0-50 | $\frac{20.8*21}{32.8} = 13.3$ | $\frac{12*21}{32.8} = 7.7$ | 21 | 21 |
| 50-100 | $\frac{4.2*34}{22.2} = 6.4$ | $\frac{18*34}{22.2} = 27.6$ | 34 | 34 |
| Current total | 19.7 | 35.3 | 55 | |
| Target total | 25 | 30 | | 55 |

Table 1.5 – Example - Iteration 1, variable 2 : the age

Note that the algorithm has some limitations for example in presence of empty cells in the sample (Lovelace et al., 2015). Agents will never be created with such combination of attributes. Moreover, the correlation can be easily damaged by the method. Finally, the final weights are decimals and not integers, implying a difficulty to transform the cross-table into a synthetic population. This problem is addressed in Lovelace and Ballas (2013), proposing to compute the number of individuals related to the entire part of the value and adapt the remainder so that we obtain the probabilities of each cell for the remaining individuals to generate. However, performing this way could results in a final population not exactly responding to all constraints.

# Part II

# Two main applications

Chapter 2

# A synthetic population of space debris

This chapter illustrates the potential of microsimulation for interdisciplinary research. Indeed, we collaborated with a team of researchers in celestial dynamics (Dr Alexis Petit, Prof. Daniel Casanova and Prof. Anne Lemaître) to generate a synthetic population of space debris, defined as no longer functional satellites or any relatively small human-made objects resulting from past spatial missions. The presence of space debris is challenging for current and future space missions because they could impact new satellites and thus prevent them from functioning optimally. This problem is still more important for the geostationary orbit, that corresponds to the orbit turning simultaneously with the Earth (about 36000 kilometres from Earth's surface). This orbit is important for satellites performing weather monitoring, phone or television communication.

We can spot and track objects with size about 1 meter in the geostationary region by optical telescope means. A network (United States Space Surveillance Network) of optical and radar telescopes is charged with detection, tracking, cataloguing and identifying of artificial objects orbiting Earth. Using the observations of the USSSN network, the USSTRATCOM (U.S. Strategic Command) produces the (pseudo-)observations "Two Line Element" each day for all objects identified, tracked, and catalogued ($\approx 20000$). They are available online on the website `www.spacetrack.org`.

However, a huge population of space debris still remains unknown, because we can observe only the brightest and biggest objects. In this context, simulations need to be implemented. There exist deterministic simulations, considering initial objects and a combination of orbit propagators, fragmentation model, and historical data. This implies having the initial objects (and their characteristics), which is a problem as

explained previously. For this reason, we propose to use microsimulation, complementary to the deterministic method.

This collaboration generated two publications (Petit et al. (2017) and Petit et al. (2018)). The first paper (Petit et al., 2017) describes first the simulation of space debris with a numerical orbit propagator and fragmentation model (described here later). Then, aware of the weaknesses of this simulation, due to missing information about some spatial debris and collisions, the Iterative Proportional Fitting is used, with the simulated debris acting as the sample and statistical estimated information (important for the small undetectable objects) acting as the constraints. The second paper (Petit et al., 2018) contains an improved version of the simulation with the use of IPF to reach two different goals : generating new space debris coming from a same distribution and reducing the discrepancies between the IPF model and the population simulated with the deterministic method. To provide the reader with an introduction to the results, the remainder of the chapter has been extracted from the second publication :

---

A. Petit, D. Casanova, M. Dumont, A. Lemaitre (2018), Creation of a synthetic population of space debris to reduce discrepancies between simulation and observations, *Celestial Mechanics and Dynamical Astronomy, Springer*, Vol 130, N 12, p. 79

---

## 2.1 Introduction

In 1957 with the launch of Sputnik I the space era began. Since that precise moment, we have been leaving behind all kinds of debris in Space. In particular, the United Nations COmmittee on the Peaceful Uses of Outer Space (UNCOPUOS) defines space debris as "all man-made objects, including fragments and elements thereof, in Earth orbit or re-entering the atmosphere, that are non-functional[(1)]". Such debris include non-functional spacecraft, abandoned launch vehicle stages, pieces of debris coming from different missions, explosions (intentional or non-intentional), collisions, satellite-surface degradation due to solar radiation or small impacts, etc.

In the last decades, several authors have dealt with space debris; from modelling the short-term evolution of space debris (Wnuk, 1996) to modelling the long-term evolution of space debris considering the solar radiation pressure (Valk et al., 2009), the shadowing effects (Hubaux and Lemaître, 2013), or short- and long-term evolution of space debris under different perturbations (Casanova et al., 2015). On the other hand, other authors focused on the global dynamics of space debris (Celletti et al., 2017). Furthermore, not only the orbital evolution is significant, but also the study of collisions between space debris and satellites (for details see Valsecchi and Rossi (2002) and Rossi and Valsecchi (2006)), how to avoid them (Casanova et al., 2014),

[(1)] Space Debris Mitigation Guidelines of the Committee on the Peaceful Uses of Outer Space. United Nations. Vienna. 2010

or a tool to quantify the catastrophic collision risk and consequences in the coming decades (Rossi et al., 2016).

However, in this work we will focus on how to give an estimation of the unknown population of space debris. Indeed, the observational means only allow us to detect the biggest and brightest objects in space and consequently, space debris which are smaller than 10 cm in Low Earth Orbit (LEO), and 1 m in Geostationary Earth Orbit (GEO), are difficult to track, and even worse to catalogue. We have to remark that, in LEO region, in situ measurements are available for sub-millimetre size objects and some statistical information is regularly acquired for objects in the 2-10 cm range (24 h radar staring experiments), which then gives strong constraints for the space debris population models. On the other hand, in GEO region, no in situ measurements are available and only sparse data on objects smaller than 1 m are available. Then, using these different sets of data, several models have been developed to model the space orbital environment and to give an estimation of the unknown population. In particular, Meteoroid and Space Debris Terrestrial Environment Reference (MASTER) (Flegel et al., 2009) and Orbital Debris Engineering Models (ORDEM) are the most popular space debris models. The evolution of the population generated is then handled by tools like LEO-to-GEO ENvironment Debris model (LEGEND) (Liou et al., 2004), Debris Analysis and Monitoring Architecture for the Geosynchronous Environment model (DAMAGE) (Lewis et al., 2001), and Semi-Deterministic Model (SMD) (Rossi et al., 2009). They are semi-deterministic models whose purpose is to model the different sources of the space debris and to propagate the orbit of each individual fragment or group of fragments.

An important issue is the comparison between space debris predictions and optical or radar ground-station observations, or in-situ measures. In particular, in the GEO region, several space debris surveys have been performed since the nineties, and they show the existence of unknown space debris populations (Schildknecht et al., 2004). Modifying the parameters of a space debris model, comparing the obtained results with the existing observations, and iterating this procedure until the simulation and the observable objects fit almost perfectly we can converge toward an accurate model. Then, as a result we have a synthetic population in agreement with the observations. Moreover, thanks to different simulations, in the GEO region the clusters of space debris observed could be explained by just eight (unconfirmed) fragmentations (Jehn et al., 2006).

The modelling of the orbital environment is a complex task. Even if we know the sources of space debris, many factors are impossible to model with accuracy. Nowadays, the NASA[(2)] Breakup Model (NBM) is useful for modelling the debris clouds generated by on-orbit explosions and collisions in terms of fragment size and velocity distributions. This model is based on a limited set of controlled terrestrial experiments like hypervelocity impacts, and data coming from the observations of clouds gener-

[(2)]National Aeronautics and Space Administration (NASA)

ated by some historical breakups (Johnson et al., 2001). The empirical nature of the NBM makes it improvable. A calibration iteration process can be applied to modify the parameters of the space debris models and to obtain a better agreement with the different sets of observational data. However, unknown events, assumptions of the source models, or limited computational resources, can produce discrepancies. We are also limited by the computational cost of propagating huge populations of space debris since we count approximately 20,000 space debris with a size greater than 10 cm, hundreds of thousands of space debris between 1 cm and 10 cm, and millions of pieces smaller than 1 mm.

The final goal of this work is the creation of a synthetic population of space debris whose global statistical characteristics are similar to the assumed as real, from the synthetic population already created by a space debris model and using additional constraints coming from different sources (for example, observations, simulations, ground-based experiments). This innovative way allows to create new pieces of space debris by using an Iterative Proportional Fitting (IPF) method, which takes into account the statistical properties of additional data such as constraints. Then, we create a new synthetic population. However, we must remark that the IPF technique is independent of the space debris model. The IPF method just modify the initial population by using simple additional constraints, without any influence on the used model.

## 2.2 Model of the space debris population in the geostationary region

We present a deterministic approach to generate an artificial population of space debris in the geostationary (GEO) region. Then, we ensure the validity of our method by verifying that the small amount of observed objects which are known, taken from the well-known Two Line Elements (TLE) catalogue, are close to the generated objects in the artificial population. After that, we use our simulated population to create a bigger one by using an Iterative Proportional Fitting (IPF) method.

### 2.2.1 The GEO region

The geostationary orbit is defined as the 1:1 gravitational resonance, where the semi-major axis is equal to 42164 km, i.e., where the orbital period of an object corresponds to one sidereal day (23h 56min 4s). The GEO protected region is defined as the ring around the Earth, which is delimited by the geostationary altitude (35,786 km) $\pm$ 200 km, and the latitude $\pm$ 15° around the equatorial plane (IADC). This region is very attractive for communication and observation satellites because they have a fixed position in the sky with respect to a ground observer.

Since the launch of the first GEO satellite in 1964 only five fragmentations are known to have occurred (at the date of 2017/10/22). The first one took place on June 25, 1978 caused by a malfunction of the battery of satellite Ekran 2 (mass 1,750 kg)

and the second fragmentation was caused by a failure of the upper stage Titan 3C Transtage (mass 2,500 kg) on February 21, 1992 (Johnson et al., 2008). A second explosion of an upper stage Titan 3C Transtage occurred on June 4, 2014, with the creation of 28 fragments in total (Cowardin et al., 2017). On January 16, 2016 an upper-stage Briz-M (mass 1,220 kg) breakup took place in a region close to the geostationary ring. It happened at an altitude of 34,866 km (at the subsatellite point 0.17° South, 223° East). Just 10 fragments were observed but several hundreds are expected (Orbital Debris Program Office, 2016*a*). A second breakup took place on June 26, 2016 when the satellite BeiDou G2 (mass 1,100 kg) underwent a fragmentation in the GEO region. The resulting orbit presents an apogee of 36,137 km, a perigee of 35,384 km, and an inclination of 4.7°. In particular, at least five fragments were observed but no one was officially catalogued (Orbital Debris Program Office, 2016*b*). All the informations about the five fragmentations are summarized in Table 2.1. More recently, a second minor breakup occurred for the Titan 3C Transtage (1969-013B) on February 28, 2018 with only one fragment catalogued (Orbital Debris Program Office, 2018), not considered in this work.

| Date | 1978/06/25 | 1992/02/21 | 2014/06/04 | 2016/01/16 | 2016/06/29 |
|---|---|---|---|---|---|
| Type | Spacecraft | Rocket-body | Rocket-body | Rocket-body | Spacecraft |
| Name | Ekran 2 | Titan Trans. 3C-5 | Titan Trans. 3C-17 | PROTON-M/BRIZ-M | BEIDOU G2 |
| ID SSN | 10365 | 3432 | 3692 | 41122 | 34779 |
| ID COSPAR | 1977-092A | 1968-081E | 1969-013B | 2015-075B | 2009-018A |
| $a$ [m] | 42,163.500 | 41,826.000 | 42,926.390 | 40,979.800 | 42,138.874 |
| $e$ | $1{,}779{\cdot}10^{-4}$ | $8{,}488{\cdot}10^{-3}$ | $1.291{\cdot}10^{-2}$ | $2.868{\cdot}10^{-2}$ | $8.934{\cdot}10^{-3}$ |
| $i$ [deg] | 0.100 | 11.900 | 8.706 | 0.174 | 4.716 |
| $\Omega$ [deg] | 78.390 | 21.802 | 313.195 | 135.143 | 61.365 |
| $\omega$ [deg] | 325.277 | 76.279 | 91.579 | 5.856 | 195.139 |
| $M$ [deg] | 78.390 | 284.560 | 269.90 | 221.106 | 164.399 |
| Mass [kg] | 1,750 | 2,500 | 2,500 | 1,220 | 1,100 |

Table 2.1 – Confirmed breakup events in the GEO region.

Currently, if we consider the TLE catalogue provided by the USSTRATCOM[(3)], and we filter the objects with a mean motion between 0.9 and 1.1 revolutions per day and whose name contain the tag DEB, we obtain 50 space debris in the GEO region at the date of October 22, 2017. In Figure 2.1, we plot the repartition of sources. We can observe that almost a half correspond to the fragmentation of the upper-stage Titan 3C and a quarter is related to individual objects. No object is related to BeiDou G2. Unfortunately, many presumed objects are non detectable for the instruments of the United States Space Surveillance Network (USSSN) and are not catalogued. The Astronomical Institute of the University of Bern (AIUB) and the International Scientific Optical Network (ISON) are both maintaining a catalogue of several hundred debris objects in GEO but those catalogues are not publicly accessible. Nevertheless, in the next subsection, we use the small sample of space debris catalogued to validate qualitatively our simulation.

[(3)] `https://www.space-track.org`

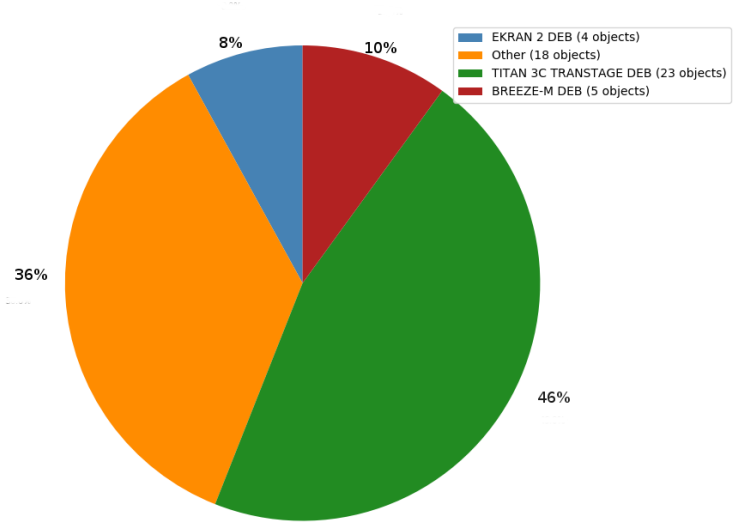


Figure 2.1 – Percentage of space debris in the geostationary region corresponding to different breakups that took place up to the date 2017/10/22. The objects gathered in the category "Other" are individuals objects not related to one of the major breakups summarized in Table 2.1.

### 2.2.2 Simulation with a deterministic method

NIMASTEP (Numerical Integration of the Motion of Artificial Satellites orbiting a TElluric Planet) is an orbit propagator (Delsate and Compère, 2012). It was written in FORTRAN[(4)], and it was developed to study the dynamics of space debris in the GEO region taking into account the geopotential, the Moon, the Sun and the solar radiation pressure with shadowing effects. The orbit of a satellite is computed by the integration of the equations of motion, expressed with Cartesian coordinates without averaging process in order to not neglect any resonant effects. An updated version of the orbit propagator includes the atmospheric drag (Petit and Lemaître, 2016). Moreover, to overcome the computing cost a parallelized hardware architecture was used allowing to propagate several orbits at the same time, hence reducing the computing time. This is particularly suited for studying large populations of space debris. In the case of the GEO region, the atmospheric drag is excluded and we prefer to use a more efficient orbit propagator named Symplec, previously developed to study the stability of a geostationary orbit perturbed by the solar radiation pressure (Hubaux et al., 2012) (Hubaux and Lemaître, 2013). Symplec is a symplectic integrator in its principles; however the motions of the Moon and of the Sun are introduced as periodic functions of time, and the energy is then periodically but not linearly, perturbed. This is why we talk about "control" and not about "conservation" of the energy. To avoid a switch

[(4)] FORmula TRANslation (FORTRAN) is a widely used programming language.

on-off in the integration, the passage through the umbra has been smoothed, using hyperbolic functions. Of course, the energy shows small periodic variations, linked to the umbra seasons, but, again, no systematic increase. Symplec code is available to work on several Central Processor Units (CPU) and it is more efficient than NIMASTEP. Thanks to this propagator, a personal computer is sufficient to propagate several tens of thousands of space debris fragments during several decades in just a couple of hours.

We performed a FORTRAN implementation of the NASA Breakup Model (NBM) following the work of Johnson et al. (2001). Given a log file with the historical fragmentations the designed implementation is able to generate a cloud of space debris at a given time and a specific location.[(5)]. The following scheme explains how the implemented code works:

Step 1: Read a file containing the initial conditions of a single object or a set of objects.

Step 2: Propagate the orbit of each object by using the software Symplec until the fixed final date or the date of the scheduled fragmentation is reached.

Step 3: Generate the initial conditions of the new fragments by calling the implemented NBM to create a larger population.

Step 4: Continue the propagation of each object until the final date or the date of the next fragmentation event is reached.

Remark that, in our study, we take into account the geopotential until order and degree five, the Sun and the Moon as third bodies, and the solar radiation pressure with shadowing effects. The simulation starts on 1978, and it ends on 2017 and we limit the minimal size of the objects at 10 cm. All scheduled fragmentations are summarized in Table 2.1.

In Figure 2.2, we plot the distribution of the fragments generated in the plane of the inclination in function of the right ascension of the ascending node and we also plot the space debris objects contained in the TLE catalogue. We observe that the generated clouds of fragments of debris are located around the known location of the TLE objects. This means that our simulation is close enough to the available data. We provide a qualitative evidence of good accordance between the simulated population and the TLE objects. However, a quantitative comparison of the generated population is a difficult task for two main reasons; the small number of objects in the TLE catalogue and the huge number of generated fragments. Nevertheless, for the purpose of this work, which is the generation of a cloud of fragments of space debris close to the TLE data, the previous simulation satisfies our expectations.

The simulated population in the GEO region agrees with available data coming from the TLE catalogue, but the rest of the population is missing from the catalogue.

[(5)] A version of the code is available on the public repository `https://gitlab.obspm.fr/apetit/nasa-breakup-model.git`

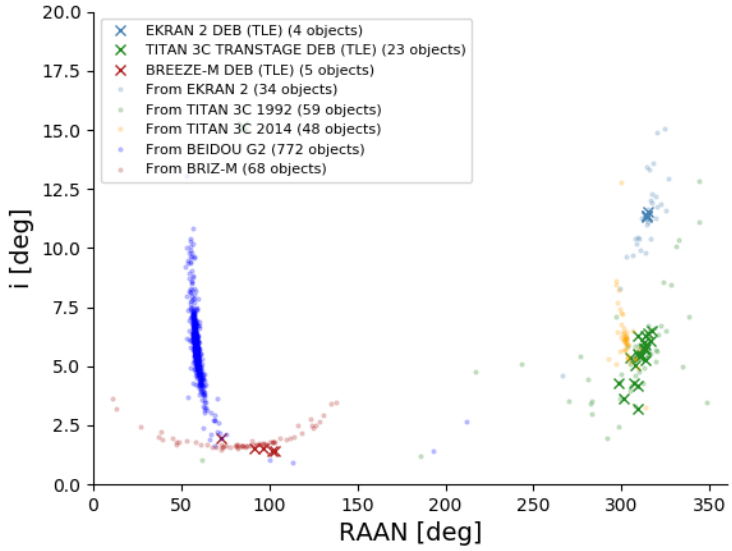


Figure 2.2 – Inclination versus Right Ascension of the Ascending Node representation of the simulated artificial population, and the 32 objects catalogued by USSTRATCOM and related to the major breakups in the GEO region. We do not consider the space debris not related to the main fragmentations.

However, the parameters of the NBM are empirical and they can not fit properly each breakup depending on a particular condition. The nominal parameters which define the distribution laws used by the NBM can not be used for each case. Even if we fit these parameters on the observations as it is the case for MASTER (Horstmann et al., 2017), the fundamental assumptions (like normal and log-normal laws used, or an isotropic distribution of the increments of velocity of the fragments) can not match the complexity of a real event.

Nevertheless, as isolated surveys give us additional statistical information, in the next section we propose a method to reduce the discrepancy between the simulation and a different set of data which comes from another simulation performed with different parameters, but in a future work, it could be replaced by real observational data. This method is not a calibration process but it allows to use data of different sources to create a new synthetic population.

## 2.3 The Iterative Proportional Fitting process

In this work, we define a synthetic population as the result of a microsimulation technique, which is the process of integrating multiple data to represent a real-world object into a consistent, accurate, and useful representation. In particular, a complete sample of data is weighted to satisfy controls using a method based on the Iterative Proportional Fitting (IPF) process, which was demonstrated by Deming and Stephan in 1940. For more explanation about the technique, we refer to Lovelace and Dumont (2016).

A piece of space debris can be located by describing its trajectory through six classical orbital elements: the semi-major axis $a$, the eccentricity $e$, the inclination $i$, the right ascension of the ascending node $\Omega$, the argument of perigee $\omega$, and the mean anomaly $M$, plus the area-to-mass ratio $\frac{A}{m}$ (we neglect other parameters such as the albedo since it produces a low impact on the global dynamics). In our work, the eccentricity, and argument of perigee are not considered to create the synthetic population since we are in a particular region were the majority of space debris move on quasi-circular orbits, according to the available data coming from the TLE catalogue. Furthermore, we are not interested in the precise position of the space debris and consequently, the mean anomaly is neither considered. However, for future works, these variables will be included to enrich the creation of the synthetic population since for objects with large $A/m$ ratio the eccentricity plays an important role for long time propagations. Thus, in our problem, we deal with four variables: $a$, $\Omega$, $i$ and $\frac{A}{m}$, which are discretized to apply the IPF process. Note that, when the different variables are independent, IPF is not necessary since each attribute of the synthetic population can be generated separately. In our case, to determine the correlation between the variables, we use the Pearson correlation coefficient, which is a measure of the linear association, between -1 and +1. If the variables are uncorrelated, this Pearson correlation approaches 0. In the case of the space debris (coming from EKRAN), Table 2.2 indicates that for each possible pair of $a$, $\Omega$ and $i$, this coefficient is higher than 0.8, indicating a high correlation (and thus dependence) between these variables. On the other side, $A/m$ seems linearly uncorrelated with each other variable. Note that by construction, the evolution of $A/m$ depends on the other considered attributes, meaning that there is a dependence even if $A/m$ presents no linear correlation with respect to the three other variables considered one by one.

| Pearson correlation | $a$ | $i$ | $\Omega$ | $A/m$ |
|---|---|---|---|---|
| $a$ | 1.000 | 0.997 | 0.831 | 0.022 |
| $i$ | | 1.000 | 0.869 | 0.013 |
| $\Omega$ | | | 1.000 | -0.049 |
| $A/m$ | | | | 1.000 |

Table 2.2 – Pearson correlation coefficients for a cloud of 460 space debris coming from the EKRAN breakup and computed with the NBM.

The way by which the variables are discretized is explained for the semi-major axis, and the other variables will be discretized in a similar way. The semi-major axis will take one of $n$ possible states ($a_1$, ..., $a_n$), each one represents a range of values. For example, and without any loss of generality, we suppose that the semi-major axis takes one of 4 possible states $a \in \{a_1, a_2, a_3, a_4\}$. This means that a piece of space debris is allocated to one region if and only if the semi-major axis has a value between the lower and upper bound (see Table 2.3) of this region. As an example, if a piece of space debris has a semi-major axis equal to 42,302 km, this means that the semi-major axis will be allocated to the state $a_3$.

| Possible state ($a$) | Interval (km) |
|---|---|
| $a_1$ | [41,000 , 41,500] |
| $a_2$ | ]41,500 , 42,000] |
| $a_3$ | ]42,000 , 42,500] |
| $a_4$ | ]42,500 , 43,000] |

Table 2.3 – Semi-major axis takes four possible states given through upper and lower bounds

Consequently, each object of the population of space debris can be allocated to a state, as in Table 2.4, depending on the possible states of each variable of the problem. We limit our analysis to $x$ objects. Each object is defined with its four variables: the first three, the semi-major axis, the inclination and the right ascension of the ascending node have for example four possible states, whereas, the fourth, the area-to-mass ratio has three possible states.

| Space Debris ID | $a$ | $i$ | $\Omega$ | $\frac{A}{m}$ |
|---|---|---|---|---|
| 1 | $a_1$ | $i_2$ | $\Omega_1$ | $\frac{A}{m}_1$ |
| 2 | $a_4$ | $i_1$ | $\Omega_1$ | $\frac{A}{m}_1$ |
| 3 | $a_1$ | $i_3$ | $\Omega_3$ | $\frac{A}{m}_2$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $x$ | $a_3$ | $i_2$ | $\Omega_4$ | $\frac{A}{m}_3$ |

Table 2.4 – Allocation of the individuals into different regions. Semi-major axis, inclination and right ascension of the ascending node are allocated in four different regions, while area-to-mass ratio is classified into three different regions.

Once we have allocated the population of space debris, to apply the IPF process, we have to build a contingency table $\Pi$, which is a matrix of dimension $n_a \times n_i \times n_\Omega \times n_{\frac{A}{m}}$, where each cell contains the initial frequencies (usually based on a sample of the population). This table is simply calculated by counting the number of objects in each possible combination of the four variables.

In this work, the contingency table is a four dimensional matrix, but for clarity, we show in Table 2.5 the contingency table $\Pi$ in a two dimensional case, taking into account only the semi-major axis and the area-to-mass ratio $\frac{A}{m}$. To better understand how to read the contingency table we give two examples; the number of objects whose semi-major axis is equal to $a_1$ and whose area-to-mass ratio is equal to $\frac{A}{m}_2$ are 4. The number of objects whose area-to-mass ratio is equal to $\frac{A}{m}_3$ is 9.

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | Total |
|---|---|---|---|---|---|
| $\frac{A}{m}_1$ | 1 | 2 | 1 | 1 | 5 |
| $\frac{A}{m}_2$ | 4 | 1 | 1 | 0 | 6 |
| $\frac{A}{m}_3$ | 1 | 3 | 3 | 2 | 9 |
| Total | 6 | 6 | 5 | 3 | 20 |

Table 2.5 – Expression of the contingency table restricted to two dimensions. The last column and the last row give the marginal frequencies. The last cell gives the total number of objects.

Before applying the IPF method we have to define constraints. In this aim, we estimate the number of objects to create in each range of values and the total number of space debris in the synthetic population (from additional data). After that, we will describe the IPF process restricted to two dimensions to facilitate the understanding.

Let $\Pi$ be the contingency table and each cell be denoted by $\Pi_{i,j}$. The marginal controls for the $i$-th row and $j$-th column are noted $m_i$ and $m_j$ respectively. The IPF process is an iterative method, which will weight the frequencies to fit the marginal controls one after the other. If we write $\Pi^t$ the contingency table at the $t$-iteration, the row-fitting is implemented as,

$$\Pi^t_{i,j} = \Pi^{t-1}_{i,j} \frac{m_i}{\sum_k \Pi^{t-1}_{i,k}} \quad \forall i, j,$$

and the column-fitting is implemented as,

$$\Pi^t_{i,j} = \Pi^{t-1}_{i,j} \frac{m_j}{\sum_k \Pi^{t-1}_{k,j}} \quad \forall i, j.$$

For example, to give the intuition behind this formula, we illustrate the process with Tables 2.6, 2.7 and 2.8, already limiting us to a contingency table with two dimensions. In Table 2.6, we add the theoretical marginal controls inferred from additional data coming from other sources, i.e. isolated observations, new simulations, statistical data, etc. In the last row and last column, we have the total number of space debris in the new population.

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | Total | Theoretical total |
|---|---|---|---|---|---|---|
| $\frac{A}{m}1$ | 1 | 2 | 1 | 1 | 5 | 10 |
| $\frac{A}{m}2$ | 4 | 1 | 1 | 0 | 6 | 12 |
| $\frac{A}{m}3$ | 1 | 3 | 3 | 2 | 9 | 11 |
| Total | 6 | 6 | 5 | 3 | 20 | |
| Theoretical total | 12 | 9 | 8 | 4 | | 33 |

Table 2.6 – Expression of the contingency table restricted to two dimensions. The current and theoretical (constrained) totals are in the last two rows and columns of the matrix.

Fitting the rows consists in re-weighting the cells to perfectly fit to the theoretical marginals of the area-to-mass ratio. For example, in Table 2.7, the cell $(1,1)$ is multiplied by 10 and divided by 5.

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | Total | Theoretical total |
|---|---|---|---|---|---|---|
| $\frac{A}{m}1$ | $1\times\frac{10}{5}$ | $2\times\frac{10}{5}$ | $1\times\frac{10}{5}$ | $1\times\frac{10}{5}$ | 5 | 10 |
| $\frac{A}{m}2$ | $4\times\frac{12}{6}$ | $1\times\frac{12}{6}$ | $1\times\frac{12}{6}$ | $0\times\frac{12}{6}$ | 6 | 12 |
| $\frac{A}{m}3$ | $1\times\frac{11}{9}$ | $3\times\frac{11}{9}$ | $3\times\frac{11}{9}$ | $2\times\frac{11}{9}$ | 9 | 11 |
| Total | 6 | 6 | 5 | 3 | 20 | |
| Theoretical total | 12 | 9 | 8 | 4 | | 33 |

Table 2.7 – Expression of the contingency table restricted to two dimensions. First iteration of IPF for the row fitting.

The first step of the first iteration still needs to update the cells and the current totals. Table 2.8 indicates that after the fitting for the area-to-mass ratio, we perfectly follow the constraint for this variable, but not for the other one. This is the reason why each iteration performs such a process for each constraint.

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | Total | Theoretical total |
|---|---|---|---|---|---|---|
| $\frac{A}{m}1$ | 2 | 4 | 2 | 2 | 10 | 10 |
| $\frac{A}{m}2$ | 8 | 2 | 2 | 0 | 12 | 12 |
| $\frac{A}{m}3$ | 1.22 | 3.67 | 3.67 | 2.44 | 11 | 11 |
| Total | 11.22 | 9.67 | 7.67 | 4.44 | 33 | |
| Theoretical total | 12 | 9 | 8 | 4 | | 33 |

Table 2.8 – Expression of the contingency table restricted to two dimensions: table after the fit of the first constraint at the first iteration.

Since we are dealing with an iterative procedure we need a stopping condition. Thus, we consider that the convergence of the method is reached when the difference between two consecutive contingency tables is close to zero (less than the machine epsilon ($10^{-16}$) for each cell difference). Thus, we compute the distance by using the following equation :

$$D(\Pi^t_{i,j}, \Pi^{t-1}_{i,j}) = \sum_{i,j} |\Pi^t_{i,j} - \Pi^{t-1}_{i,j}|.$$

In particular, we fix this stopping condition to $10^{-13}$ when the chosen discretization produces a contingency table of 960 cells ($960x10^{-16} \approx 10^{-13}$).

In Figure 2.3, we show in a flowchart the followed process to apply the IPF method. We see that two sets of data are used: the data coming from the simulations (at the left), and the additional data used as constraints (at the right). The first one gives the cross-table with the frequencies associated, and the second one gives the constraints of the new population. Then, we apply the IPF process to create the new cross-table, which will be converted in a new population of objects. Nevertheless, the created cross-table by the IPF process contains real values, but integers are required. We apply the Truncate, Replicate and Sample (TRS) method (Lovelace and Ballas, 2013). First, we truncate each cell of the contingency table to keep only the integer part. Then, the total population is smaller than the target population. We have to add fragments in the contingency table to complete the population. For this purpose, the decimal part is kept in a weight table, and normalized to obtain the sum of the weights equal to one.Then, the probability to add a fragment is given by this weight table. The missing fragments are chosen by following these probabilities.

The final part consists of a conversion of the new contingency table in the form of Table 2.3, i.e in a list of fragments whose variables are defined by a class. To obtain real values, we compute a random value in the bounds of the class, following the distribution law used to determine the constraint.

## 2.4 Applications

Once we have simulated a population of space debris in the GEO region in a deterministic way considering all the breakup events that took place in the last decades, and once we have explained how the Iterative Proportional Fitting (IPF) method works, we can apply both tools together. The first application consists of applying the IPF method to create a bigger population of space debris by using as constraints the inferred data from the simulated population in Section 2.2.2. Then, the second application consists of analysing how constraints will influence the creation of synthetic populations.

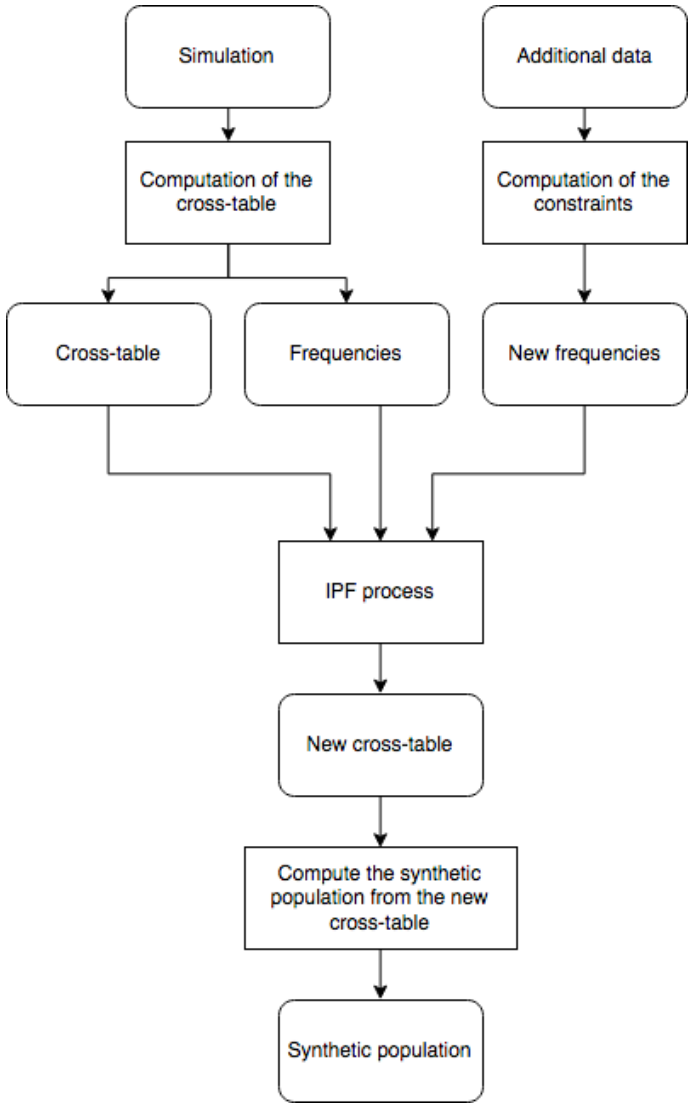


Figure 2.3 – Flowchart of the IPF process.

### 2.4.1 First application: generation of new space debris

Without loss of generality, we propose to focus the study on the cloud created by the breakup of Ekran 2. The IPF method will modify the population simulated according to the inferred constraints of the considered population. The final purpose is to validate the IPF method, but also to use this methodology to create a reliable population from a small sample, and save computational time.

In Figure 2.4, we plot the distribution of each variable obtained by the selection of the 460 fragments (with a minimum size of 1 cm) of the cloud created by the breakup of Ekran 2, and propagated until the date of October 16, 2016.

Taking into account the fact that the orbits of the fragments differ from the parent body due to the isotropic velocity increment, we can assume the distributions of the variables $a$, $i$ and $\omega$ follow Cauchy laws centred around the mean values (a Kolmogorov-Smirnov confirms this assumption with $p$-values always greater than 0.4). We keep in mind that this is not true for all cases and it depends on the dynamics. Then, a more complex distribution law (or the empirical distribution) could fit in a better way, however, we consider that the Cauchy law is well suited and useful for showing the proposed methodology. For the $\frac{A}{m}$ ratio we use a lognormal law since the NBM uses this kind of distribution. Furthermore, in Figure 2.4 we observe that the selected distributions fit perfectly with the available data. These distributions are used to compute the new frequencies of the synthetic population by using a Monte-Carlo method and fixing the total number of objects of this bigger population. In this example, we double the number of fragments of the initial population.

In order to illustrate the convergence of the IPF method, we plot in Figure 2.5 the distance computed with equation (2.3) at each iteration. We observe that the IPF method converges in less than 100 iterations and it stops when a minimal distance equal to $10^{-13}$ is reached. Note that the computation is fast and takes just several seconds.

Finally, in Figure 2.6, we show a comparison between two families of objects; the ones corresponding to the simulated population (460 objects), corresponding to the ones illustrated in Figure 2.2, and the ones corresponding to the synthetic population, created thanks to the IPF method (920 objects). Keep in mind that we do not compare the entire population, we focus on a particular region, i.e. objects whose right ascension of the ascending node is in a range 300-330°, and whose inclination is in the range 8-16°. Figure 2.6 indicates that the synthetic population seems to be located in the same region as the population in terms of inclination and RAAN. The Pearson correlation coefficient between RAAN and inclination is 0.75, meaning that the IPF has conserved a high positive correlation between these variables. However, the simulation showed a slightly higher correlation (0.869 as seen in Table 2.2). A hypothesis test of Kolmogorov-Smirnov gives a $p$-value greater than 0.4 when testing if inclination and RAAN follows the same Cauchy distribution as initially. That means, as Figure 2.7 shows, that the synthetic population follows the assumed distribution.

In conclusion, for this application, the method succeeded in creating a larger population of space debris by keeping high correlations and following the distributions used as constraints.

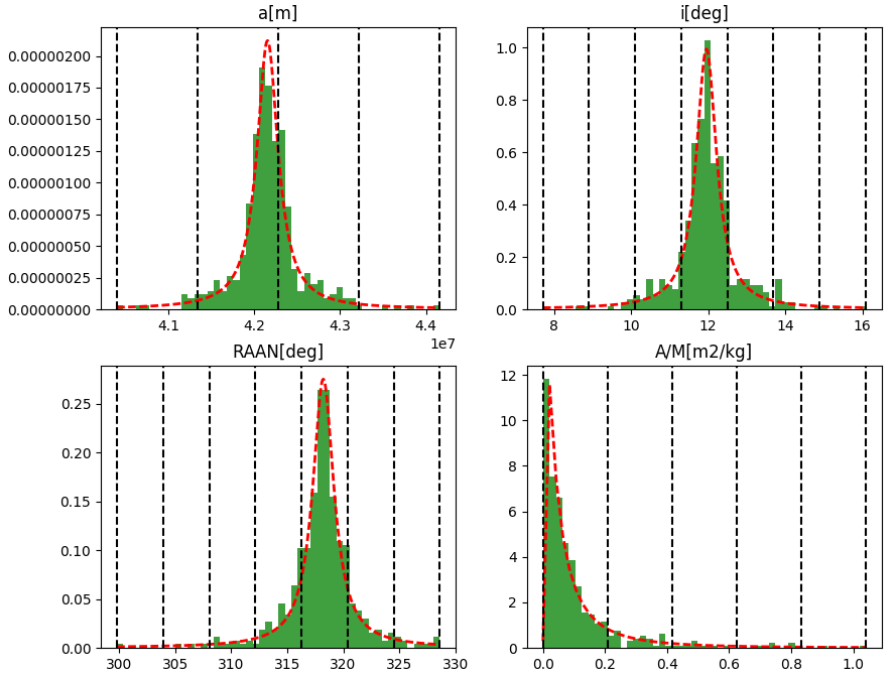


Figure 2.4 – The distribution of the variables $a$, $i$, $\Omega$, $\frac{A}{m}$ of the simulation and the fit by a Cauchy law for the first three ones and by a lognormal law for the last one.

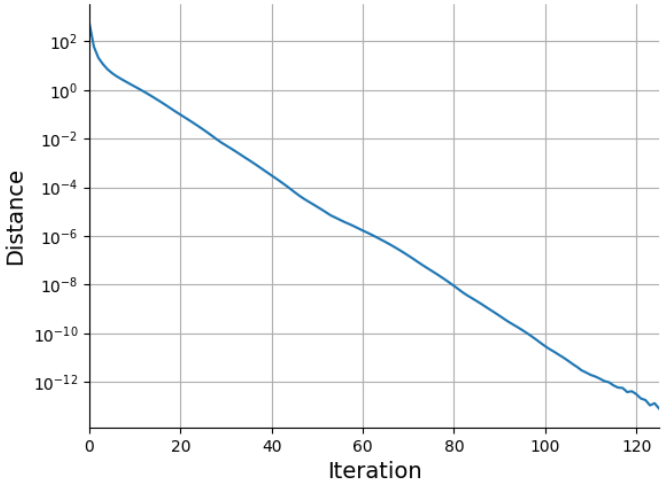


Figure 2.5 – Evolution of the distance computed with the equation (2.3) at each iteration.

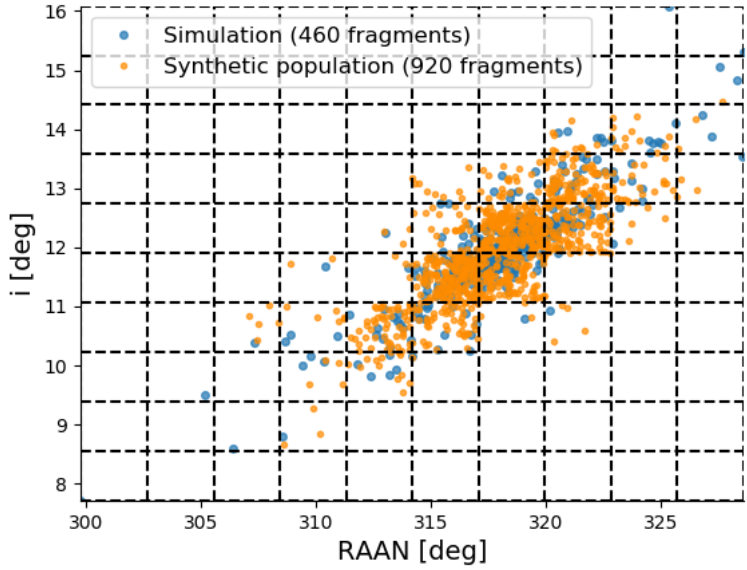


Figure 2.6 – Comparison between the simulation and the synthetic population.

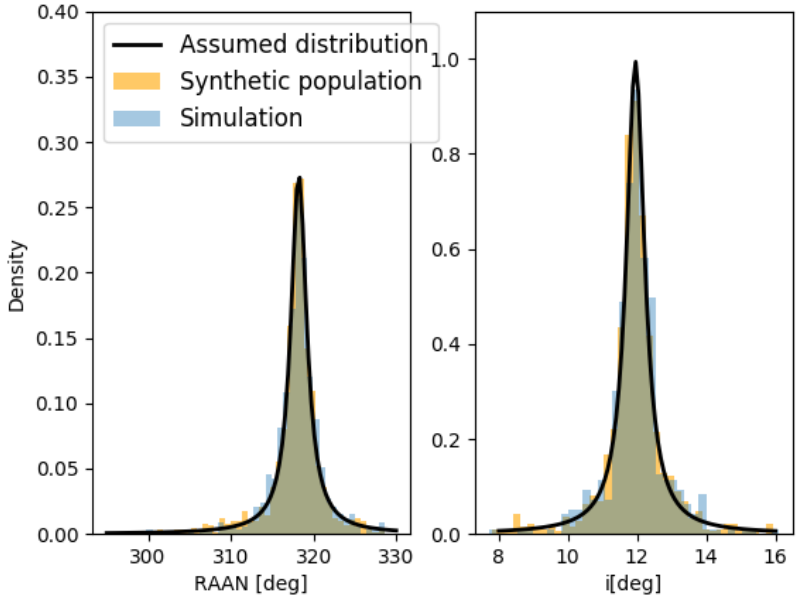


Figure 2.7 – Distribution of RAAN and inclination for the simulation and the synthetic population.

### 2.4.2 Second application: reducing discrepancies between IPF model and the simulated population

In this second application we consider again the case of Ekran 2 population, but this time constraints differ from the population simulated in Section 2.2. Indeed, we alter the NASA Breakup Model (NBM) and we obtain a different cloud of space debris in the simulation. The fragments of this cloud have different distributions and consequently, we infer different constraints, with which the IPF method will produce a different synthetic populations. Thus, we can observe how different constraints in the initial simulation can influence the creation of the synthetic population.

The NBM gives an increment of velocity $\Delta V^{NBM}$ for each fragment, but in the reality, the ejection velocity of fragments will depend on the energy of the event and then, when the cause is unknown, it is impossible to estimate the ejection velocity. We will consider two different cases; one of them with high ejection velocities of the fragments, and a second case with low ejection velocities. For this purpose, we introduce a factor $\beta$ to obtain the increment of modified velocity $\Delta V^{modif} = \beta\ \Delta V^{NBM}$.

We assume that the ejection velocities of the fragments produced by the explosion of the satellite Ekran 2 are ten times smaller than the ejection velocities considered in the simulation presented in Section 2.2, i.e., we use $\beta = \frac{1}{10}$. Moreover, we take only the set of debris with a characteristic size above 1 cm. In Figure 2.8, we compare the distribution of the cloud produced with the nominal increments of velocity (simulation presented in Section 2.2) and the cloud produced with the modified ones. We observe that the second cloud (simulation 2) appears less expanded in the considered plane.

We keep the first simulation produced with the nominal values of the NASA breakup model as our initial population. We apply the IPF method using new constraints computed with the second simulation, where the increments of velocity were divided by ten. Note that this is an extreme test for the method. Indeed, the second simulation does not follow the same correlation as the first one.

The synthetic population created is compared with the population of the first simulation in Figure 2.9. We can observe that, as desired, the shape of the scatter plot has been changed by the procedure. However, even if this change is in the direction suggested, the synthetic population does not really follow the tendency of simulation 2 used as constraint (shown in Figure 2.8). As seen on Figure 2.8, a high linear correlation is present for both simulations (0.869 for simulation 1 and 0.910 for simulation 2), but not following the same relation. This confuses the IPF process and the synthetic population has a correlation of only 0.529. Thus, the positive correlation is still present but less evident. This is caused by the dense "square" of dots around a RAAN value of 318 degrees that we can observe on Figure 2.8.

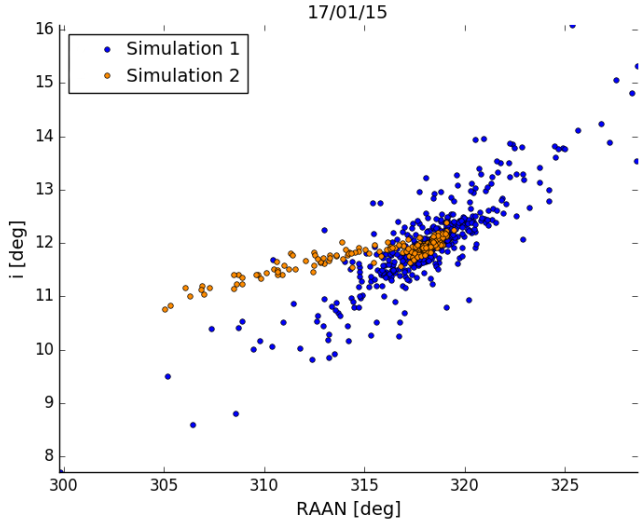


Figure 2.8 – The first simulation of the Ekran 2 cloud is performed with the nominal increments of velocity. The second simulation is performed dividing by ten the increments of velocity.

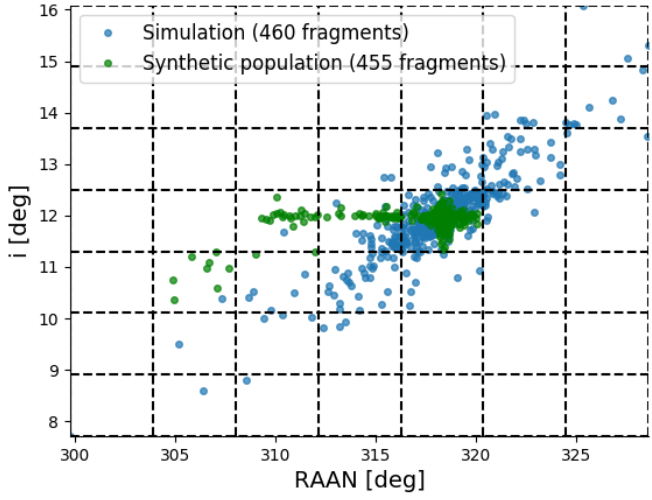


Figure 2.9 – Comparison between the nominal simulation and the synthetic population created with the modified simulation.

Figure 2.10 contains the two simulations, the synthetic population (created from the constraints of the second simulation) and the linear regressions associated with each set of space debris. The regression lines of the simulation 2 and the synthetic population stand close together in comparison to simulation 1, indicating a good (but not perfect) improvement of the simulation with the IPF.

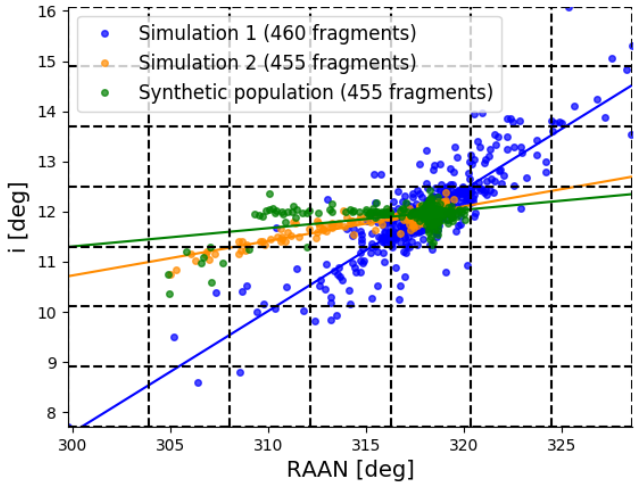


Figure 2.10 – Comparison between the two simulations and the synthetic population created with the modified simulation.

Deduce a simulation from another one thanks to microsimulation is a challenging task. It is possible to summarize the followed procedure:

Step 1: Construct discretized constraints after a continuous fit of the distribution for simulation 2.

Step 2: Run an IPF initialized to simulation 1 with the constraints defined before. This step gives the number of object per discretized zone (the square on the graph).

Step 3: Create a population of space debris by determining for each object a specific attribute for each variables (thanks to the known cell that gives the range, and thanks to the continuous distribution of each variable).

This section shows that this method is quite satisfactory and we could improve by adapting the first and third steps to the application. More precisely, we have several propositions for a future work.

- Try to use as constraint a density function adapted to the simulation (in this work we just took a log-normal distribution for $A/m$ and a Cauchy distribution for all other variables).

- Discretize the environment, i.e. a grid of cells of same dimension is used in this work, but to use a grid based on the quantiles could give better results.
- Tackle the zero cell problem of the IPF method, i.e. when the target simulation needs individuals in a cell not present in the simulation, they will never be in the synthetic population. This zero cell problem can be avoid easily by adapting the size of the intervals or by replacing the zeros by a very small value as explained in Suesse et al. (2017).

Note that step three of the explained procedure is a stochastic process, meaning that running the code several times could give slightly different results.

## 2.5 Conclusion

In this chapter we first present a deterministic approach to generate an artificial population of space debris in the geostationary (GEO) region in agreement with the population provided by the USSTRATCOM catalogue. Then, we use that generated population to create a new one by using a microsimulation method (IPF technique). This method is based on a process of integrating multiple data to represent a real-word object into a consistent, accurate, and useful representation including in the model additional constraints. The purpose is to create a population of space debris whose global characteristics are closer to a population assumed as real since it merges from different sets of observational data used as constraints. Indeed, the deterministic nature of the simulations produced by the Breakup Model could give large discrepancies between the simulated space debris population (with a space debris model) and the observed space debris population, since the initial conditions totally determine the whole trajectory, without trying to fit to new observational data. Furthermore, the limitations of our knowledge about the events occurred in orbit, the assumptions made in the source models, the limited computational resources, limit the improvements of the calibration iterative process of the model parameters in order to obtain a space debris model in accordance with observations.

In this work we show how to create a synthetic population of space debris by using an IPF technique. We provide two applications. The first one consists of creating a synthetic population with the same statistical properties than the assumed as real, but it includes more objects; in this application the new pieces are inferred from a similar set of data. The second application provides a synthetic population different from the initial one. In this case, the final goal is to show the influence of the statistical properties used as constraints in the creation of a synthetic population of space debris.

This model is a first step of producing a synthetic population of space debris, which explains the main idea of the process and shows the relevance of this method for the space debris models. This chapter thus showed that the theories concerning spatial microsimulation are really adaptable to problem in other field than demography. The method presented in this chapter differs from usual demographic simulations

in the sense that no exhaustive data exists, whereas in demography, census are often used containing information on the whole population, or at least the total number of individuals. Note that the usual deterministic models used in celestial mechanics are based on assumptions on collisions and when the parameters are fixed, the involved laws completely determine the whole simulation. In these kind of models, the observational data can be used to calibrate the initial parameters, but unpredicted events in the trajectory (collisions with unknown objects for example) implies potentially big differences between the observation and the simulation. Estimating the space debris localisation with microsimulation techniques is very promising, since it doesn't need to know the history of each space debris and aims at fitting well the observational data. These kind of hybrid models using deterministic simulations and microsimulation together deserve to be more investigated.

Chapter 3

# Virtual Belgium In Health

This chapter covers the results of a research funded by the Walloon Region conducted from 2014 to 2017, in the program WB-Health[(1)]. Previously a model to simulate the population of Belgium aiming at predicting the traffic demand (Barthélemy, 2014) has been developed in the University of Namur. Starting from these premises, the Walloon Region project was born to perform similar forecasting simulations, but focusing on the health of the elderly population. A version of the final developed interactive software has been delivered in 2017, allowing the DGO5 to perform simulations by adding assumptions such as statistical information about a disease per age, gender and municipality for example. Note that the research reported in this part of the thesis was subjected to a precise timetable, restraining considerably the time to test and compare different methods for each task, since the RW was more interested in the final product than in the annexed research questions.

## 3.1 Introduction

Nowadays, it is important for the decision makers to be aware of the current population and to have an idea of the evolution of this population through time and space. With this kind of information, they can better organize the society and analyse the different needs. However, this knowledge is difficult to acquire due to privacy concerns and also due to the lack of disaggregated and crossed data. Therefore microsimulation techniques have to be used for these purposes. A commonly employed methodology is the synthetic population building (Cho et al. (n.d.) and Ballas and Clarke (2001)). For example, in the domain of transport research, there are many cases where synthetic populations have been created (Barthélemy, 2014).

[(1)]More information about this contract can be found here : `https://recherche-technologie.wallonie.be/projets/index.html?IDD=25044`

The ultimate goal of this research is to study the present and foresee the future (until year 2030) health needs for the ageing Belgian population. Aware of the local differences and the impact of the household's structure, the project involves a microsimulation, including individual attributes. Moreover the individuals are also grouped into households and localised in a specific municipality, since the most disaggregated statistics available is at municipality level. To simulate the time evolution of the population, a starting population and information of the way it evolves are needed. This chapter is thus split into several sections, each one regarding one specific step of the research. First, a static microsimulation is performed, resulting in a static synthetic population for 2011, the first of January, ready for the evolution. Then, the time evolution is modelled by means of specific dynamical processes, with a fixed timestep of one year. Finally, the results are analysed and the perspectives and limitations of the framework are discussed.

## 3.2 Initial static synthetic population

Since the last census available for Belgium is the administrative census executed in 2011, we decided to create the static initial Belgian population for this year. The early process described in this section has been published in: Dumont et al. (2017*c*). Our previous results have been repeated to get better estimates on the errors.

Our goal is to obtain an initial Belgian population for 2011. However, due to privacy concerns, we can not access the exhaustive records from the whole National Register. Even if aggregated contingency tables could be provided, we may not receive neither a listing containing all characteristics of all Belgian inhabitants, nor the cross-tabulated counts of all combinations of modalities for all variables. For this reason, we need to generate a synthetic population, which is statistically as close as possible to the actual population (according to the data available). This process will provide us with synthetic households (note that a household is defined as a set of individuals living together) and information on each households' member. This dataset will mimic the Belgian population but without any privacy concerns. Our aim is to proceed per municipality scale to be able to produce a forecast as disaggregated as possible. Indeed it is important for the authorities to know with a fine spatial mesh where the health needs occur to be able to locate as accurately as possible the health supply.

Initially, the project was planned to extend the tool VirtualBelgium (Barthélemy, 2014) so as to handle health concerns. Indeed, also for VirtualBelgium, a synthetic population has been created (Barthélemy and Toint, 2013) taking into account the whole Belgium. The innovativeness of the method is to build the synthetic population without needing a sample. Indeed the lack of a sample was the lock to pull out since only aggregated data per zones are available. The method performs in three phases :

1. Generation of a pool of individuals

2. Generation of a set of households (without individuals associated)

3. Fill the households with members from the pool of individuals.

Unfortunately, the static initial population of VirtualBelgium is calibrated with data from 2001 and some issues have been observed such as several couples with a huge age difference for example. Moreover, more precise and recent data are accessible for VBIH, allowing a better calibration of the households. However, let us observe that the limitations of the initial population of VirtualBelgium doesn't really affect the focus of VirtualBelgium, which is the simulations of the daily traffic, on the other hand such limitations are problematic once the focus of the research is on people health. Indeed, the households are crucial in this study since elderlies need later external help if they live with their children for example. For this reason, we develop a specific and more adequate tool without needs for sample data. The method we propose is an adaptation of the process performed in (Barthélemy and Toint, 2013). The global idea is just adapted, allowing to consider additional data on the households' structure. Moreover, in the individual pool, attributes such as the size and type of household and the role of the individual in the household are gathered.

Our method proceeds in two steps as illustrated in Figure 3.1. First we create a pool of individuals with a series of characteristics (the target attributes and information about their household). Then, we group these individuals into households using information about ages differences inside the households and the individual attributes about its household.

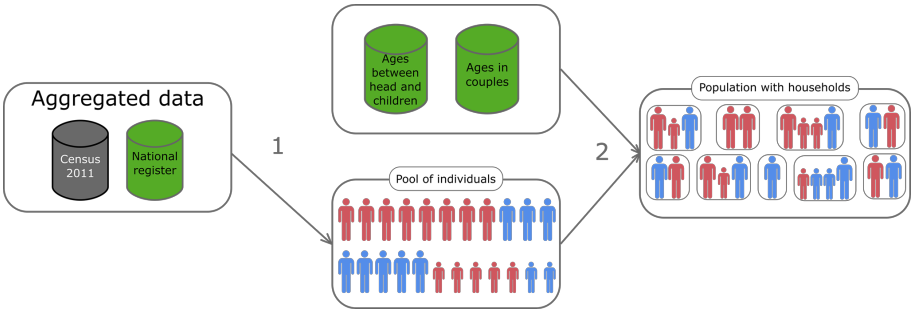


Figure 3.1 – Schematic representation of the static synthetic population creation

It is essential to mention that no perfect synthetic populations exist since they remain models of the actual population and their quality depends on the quality of the data sources, the quality of the methods, the structure of the data (correlations, etc).

### 3.2.1 Pool of individuals

To create the set of individuals, we first need to gather relevant and spatially disaggregated data. Then, we need to consider all the available data to build a population as close as possible to the actual one. At the end of this process, a pool of individuals is created, which contains for each member, the following attributes (useful later):

- municipality (INS code);
- gender (Male or female);
- age class (Per 5 years);
- civil status (Single, married, widower or other);
- professional status (Worker, unemployed or inactive) ;
- highest educational level (from the International Standard Classification of Education 1997 (ISCED 97) [(2)]):
  - No certification nor diploma
  - Primary level of education (CITE 1)
  - Lower secondary level of education (CITE2)
  - Upper secondary level of education (CITE3)
  - Post-secondary, non-tertiary education (CITE 4)
  - First stage of tertiary education (CITE 5)
  - Second stage of tertiary education (leading to an advanced research qualification) (CITE 6)
  - Not indicated (for person being more than 15 years old)
  - Not concerned (for person being less than 15 years old)
- role in the household (= link with the head):
  - Head
  - Partner
  - Child
  - Son-in-law or daughter-in-law
  - Grandchild
  - Mother or father
  - Grandmother or grandfather
  - Sister or brother

[(2)] See `https://ec.europa.eu/eurostat/cache/metadata/Annexes/educ_uoe_h_esms_an2.htm`

  - Other related
  - Without relation
  - Great grandchild
  - Uncle or aunt
  - Nephew or niece
  - Cousin

- number of members in the household (from 1 to "6 and more");
- type of household (married couple with or without children, isolated, cohabitants with or without children, monoparental family, other private household or collective household)

Two resources are mobilised at this stage. First, for this research, we collaborate with the group DEMO of UCLouvain, that has authorized access to the National Register. Nonetheless they are not allowed to provide us with the whole dataset of disaggregated variables. The most precise table which could be provided contains the number of individuals in each category crossing modalities from these attributes:

- municipality;
- gender;
- age class (per 5 years);
- civil status;
- role in the household (head, spouse, etc);
- number of members in the household;
- type of household (married couple with or without children, etc). This attribute is deduced from the different roles observed in the household, meaning, for example, that a child can be more than 18 years old.

Such a table already contains enough information to build the synthetic individuals with all the needed attributes, except the highest level of education and the professional status. Indeed, these information are not included in the National Register. Therefore we need to rely on another data source to include them.

This second source is the administrative census from 2011[(3)], that contains many variables. However, once more, the privacy issues restrain our access to only 4 crossed variables at a time. For this reason, we focus on the education level and professional status and try to choose several combinations of 4 variables to have the necessary information for the assignment of the two missing attributes (education and professional

[(3)]Databases from "Statistics Belgium", available on the website http://www.census2011.be

status). Table 3.1 summarises the chosen databases. This choice was guided by the following considerations. First, it is important to have the marginals of both variables per municipality. Moreover, the age and gender are obviously correlated with these attributes, explaining the choice of both first tables. Then, we would like to have a table that takes into account the relation between the level of education and the professional status, which are intuitively correlated. To know which two other variables to take in addition, we tested several possibilities and analysed the correlations. It appears that age and gender influence more the relation between the two target variables than the municipality. This explains the choice of the third table.

| Characteristic | Database 1 | Database 2 | Database 3 |
|---|---|---|---|
| Municipality | × | × | |
| Age class (per 5) | × | × | × |
| Gender | × | × | × |
| Education level | × | | × |
| Professional status | | × | × |

Table 3.1 – the chosen databases from the Census

Thus, a total of four contingency tables coming from two different sources are now available for our objective. With these databases, we are able to form a pool of individuals with all necessary attributes, including those needed to form the households.

To create the pool of individuals, we take the dataset from the National Register and we add the highest educational level and the professional status to each inhabitant. For this, we first rescale the tables coming from the census. This is necessary, since the census also consider the asylum seekers, while the National Register does not include them. We now have, thanks to the database 1 of the census, the number of individuals to insert in each education level per municipality, age class and gender. This allows us to draw an education level for each person out of this distribution.

Then, we try to make a similar reasoning for the activity status. However, this method does not consider the relationship between the educational level and the activity status. Thus, the results are not acceptable. This is why we choose to apply an Iterative Proportional Fitting (IPF) (Lovelace and Dumont, 2016) on the three census tables. The considered seed (the initial weights - corresponding to the sample when available) is the table containing the relationship between both variables, the age and gender at national level. The two remaining tables are established as the constraints[(4)]. The resulting table is the distribution used to generate an activity status for each individual, depending on his/her age, gender, municipality and education level.

[(4)]To execute this algorithm, we used the mipfp package of R (Barthélemy and Suesse, 2015).

To visualize the quality of this process, for each database, we compare the number of individuals in the census to the number of simulated individuals. This comparison is achieved category per category. Figure 3.2 and 3.3 illustrate this comparison for both databases linked to the municipality but without considering the interaction between the status and the diploma level. Each dot corresponds to a category and its coordinates are (number of simulated, number of desired) individuals within this category. We observe that, respectively, the simulated diploma and activity status fit both constraint tables. Indeed, the dots follow the identity function, meaning that the simulated and desired numbers are approximately identical. By analysing the fit, we observed 0.27% of individuals in the wrong crossed category of education level per municipality, age and gender and 0.34% for the crossed category of activity status per municipality, age and gender.

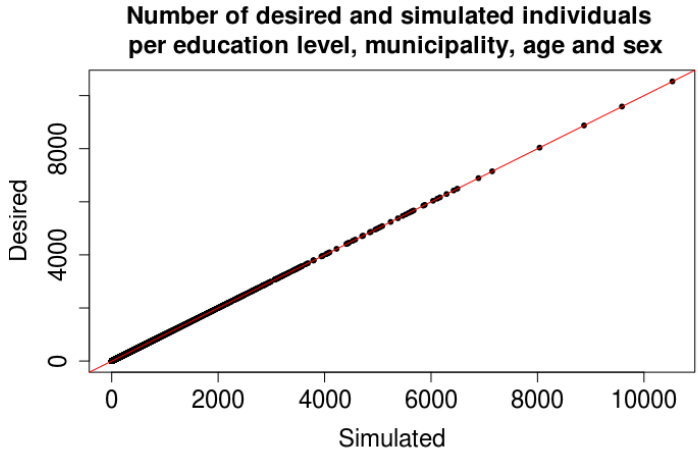


Figure 3.2 – Quality of the simulation in terms of the census database 1 : education level per municipality, age and gender

These good fitted results were expected, since these two tables were the constraints of the IPF, which has converged. The challenge is to have a final population following at least the trends of the relationship between the two attributes. Figure 3.4 shows that the fit is not perfect for this table, but globally, no category seems to be really apart from the desired number. The proportion of individuals in the wrong category of activity status per education level, age and gender is 2.45%. This table doesn't consider the municipality, meaning that the small differences are spread in the different municipalities. Moreover, it has been tested to consider this table as constraint and one of the two other tables as initial weight, but the results imply a poorer fit.

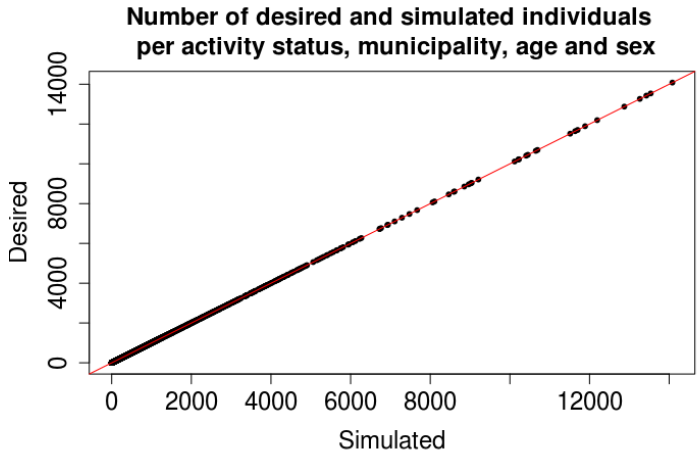


Figure 3.3 – Quality of the simulation in terms of the census database 2: activity status per municipality, age and gender

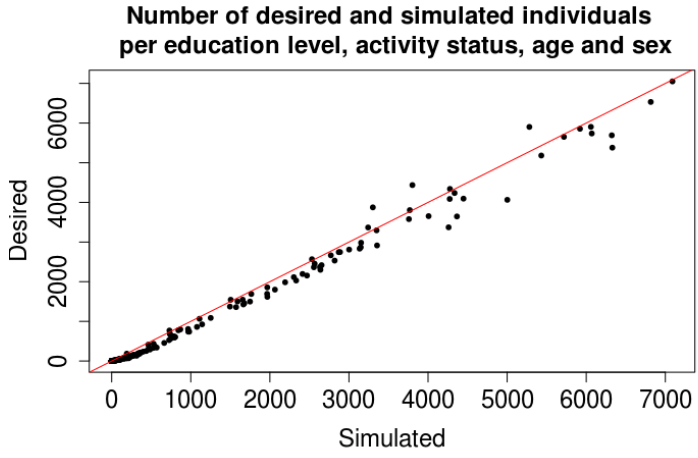


Figure 3.4 – Quality of the simulation in terms of the census database 3: activity status per education level, age and gender

We have now synthetic individuals with all target characteristics. The next step is to group these people into households.

### 3.2.2 Grouping into households

The method to group individuals into households will depend on the type of household. Indeed, we have :

- Isolated individuals (creating a household alone)
- Couples without children (needing to associate husband and spouse)
- Couples with children (needing to associate husband, spouse and children)
- Monoparental families (needing to associate children to the head of the household)
- Cohabitant without children (needing to associate the cohabitant)
- Cohabitant with children (needing to associate the cohabitants and the children)
- Other private households and Collective households

For the last categories, the "collective" households includes all individuals known to live in a retirement home for example, or any collective households. The "other households" category includes the persons not identified to another type of private households. To have an idea of the repartition, Table 3.2 contains the number of individuals registered in the National Register for 2011 per size and type of household.

| Type \ Size | 1 | 2 | 3 | 4 | 5 | 6+ |
|---|---|---|---|---|---|---|
| Isolated woman | 804.564 | 0 | 0 | 0 | 0 | 0 |
| Isolated Man | 710.448 | 0 | 0 | 0 | 0 | 0 |
| Couple without children | 0 | 1.860.372 | 0 | 0 | 0 | 0 |
| Cohabitants without children | 0 | 580.954 | 0 | 0 | 0 | 0 |
| Couple with children | 0 | 0 | 1.200.501 | 1.681.332 | 764.930 | 351.476 |
| Cohabitants with children | 0 | 0 | 390.987 | 376.064 | 117.560 | 49.967 |
| Mono parental with mum | 0 | 426.238 | 311.199 | 128.632 | 41.790 | 18.215 |
| Mono parental with dad | 0 | 98.204 | 50.538 | 17.636 | 4.555 | 1.930 |
| Collective | 0 | 0 | 1.752 | 1.788 | 1.725 | 130.386 |
| Other | 2.311 | 86.078 | 165.030 | 177.284 | 165.990 | 230.830 |

Table 3.2 – Counts of individuals per household size and type in 2011

To group the individuals into households, we consider subsets of individuals susceptible to live with each other. Thus, we split the individuals living in different types of households and in households of different sizes. For example, as illustrated on Figure 3.5, all individuals living in a household 'Married couples with children' are joined together. Then, inside this type, we divide individuals according to their household size. This means that we have now subsets inside which we have to choose who live with whom, depending on their role in the household (i.e. the link with the head). For instance, to create the households of our example for a size of four, we always need to join a head with a partner of opposite gender (in the National Register of 2011, this type of households means couples of different genders) and two children.

In this example, Head 1 is a woman and married with the partner 2, who is a man. They have two children : the 2 and 5.

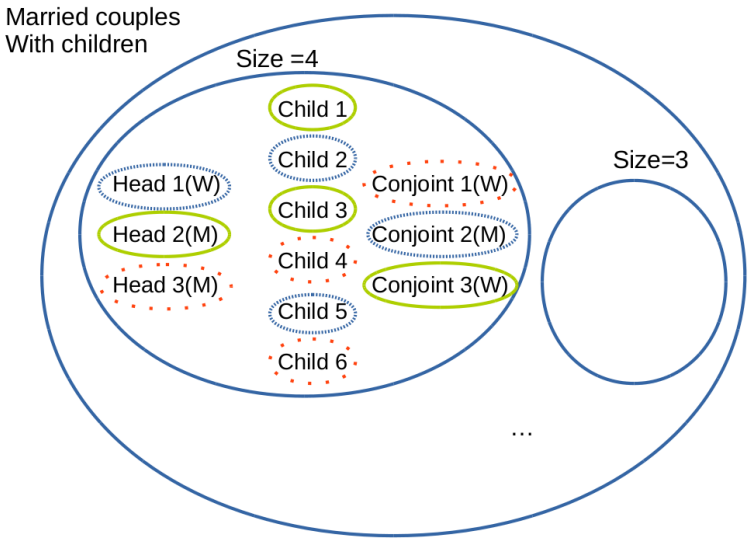


Figure 3.5 – Subset of individuals susceptible to be in the same household (same type and size of household)

To chose which head is married to which partner and who are their children, having no additional data, we can only draw the households members randomly. However, this could form some improbable couples, joining a woman of 15-20 years old and a man of 85-90 years old, which is a rather infrequent case. For this reason, the group DEMO of UCL provides us the contingency tables of the age of the woman depending on the age of the man, for all couples, per municipality and type of households. A similar data set gives the age differences between the head of the household and his/her children, for each different type of household, per municipality.

This task is a perfect example of combinatorial optimisation problems, with the distributions of age differences as constraints. Interested in the whole Belgian population, testing all possible household's grouping of the individuals to choose the one fitting the best the constraints is unthinkable. For this kind of challenges, a stochastic search can be implemented and several methods are available. Since our task here concerns the grouping into households of the total population of Belgium, a potential "candidate" would be a specific grouping. Two big categories of method exist : methods considering a "population of candidates", requiring here to record simultaneously many potential grouping of the individuals into households, whereas the other method works on one "candidate" at a time and move from one to another candidate, needing only two potential candidates recorder simultaneously. In our case study with already memory expensive candidate and since the project is requiring to minimise the computational time, the second type of method is preferred. This is why we prefer a Simulated Annealing (Hoos and Stützle, 2005) to a genetic algorithm (Goldberg, 1989). In our case, the size of household and the role inside this household must be respected, this is a constraint; whereas the age differences distributions intervene in the objective function.

The Simulated Annealing has been developed but is, in our case study here, also really computationally time consuming and optimising the parameters is a real and established challenge. Moreover, simple models are always preferred to complex models when resulting in similar simulation quality. Thus, a more basic strategy of random draws in the age difference distribution has also been tested. Both methods are described in this section. The research aiming at finding in a fixed time a reasonable synthetic population to evolve, the Simulated Annealing has been aborted when realising the necessary time to make it perform better than the random method.

#### 3.2.2.1 Simulated Annealing

Simulation Annealing, first introduced by Kirkpatrick et al. (1983), performs in two steps : create a starting candidate with random combinations and then iterate to propose an adapted version of this candidate depending on different parameters (Hoos and Stützle, 2005). At the end of each iteration, only one out of the two candidates is retained and the next iteration begins. In our case, the random first candidate already respects the constraints, namely the size of households are not mixed and the role in the household and type of household are respected. The randomness appears only to choose which individual with the good characteristics we take.

The algorithm performs iterations until a stopping criteria is met. Each iteration consists in proposing a candidate in the neighbourhood of the actual candidate and choose which of them continues the process. In our case, we define "a neighbourhood" candidate as the same population with just exchanged individuals (keeping same role, type and size of household). To choose the victorious candidate, criteria need to be decided, helped by an objective function. If the new candidate better fits the objective

function, it is retained. However, if the new candidate is worst in terms of the objective function, he still could be kept with a certain probability (decreasing with the iterations increasing). This allow to better explore the feasible set and not stay stuck in a local optimum. The rhythm of the "decreasing probability" is captured in the notion of "temperature". This step is iterated until the population stagnates (in terms of the objective function) or if the maximum number of iterations is reached.

This promising method has been implemented for the couples without children within the municipality of Namur. In this case, only one objective has to be respected: the age differences in the couples. The desired associated cross table of ages between the partners is in Table 3.3.

| M\F | 15.19 | 20.24 | 25.29 | 30.34 | 35.39 | 40.44 | 45.49 | 50.54 | 55.59 | 60.64 | 65.69 | 70.74 | 75.79 | 80.84 | 85.89 | 90.94 | 95. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15.19 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20.24 | 0 | 20 | 20 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 25.29 | 3 | 55 | 135 | 38 | 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30.34 | 1 | 11 | 89 | 61 | 15 | 5 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 35.39 | 0 | 6 | 25 | 43 | 33 | 12 | 2 | 2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 40.44 | 1 | 2 | 5 | 19 | 24 | 39 | 16 | 3 | 5 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 45.49 | 0 | 3 | 1 | 5 | 16 | 44 | 50 | 32 | 9 | 7 | 5 | 0 | 0 | 0 | 0 | 0 | 0 |
| 50.54 | 0 | 1 | 0 | 5 | 7 | 28 | 80 | 212 | 59 | 23 | 5 | 4 | 0 | 2 | 0 | 0 | 0 |
| 55.59 | 0 | 1 | 0 | 1 | 5 | 15 | 39 | 238 | 502 | 130 | 14 | 8 | 1 | 3 | 0 | 0 | 0 |
| 60.64 | 0 | 0 | 1 | 2 | 2 | 7 | 19 | 64 | 438 | 808 | 118 | 23 | 1 | 1 | 0 | 0 | 0 |
| 65.69 | 0 | 0 | 1 | 0 | 0 | 1 | 6 | 16 | 92 | 453 | 461 | 67 | 12 | 2 | 0 | 0 | 0 |
| 70.74 | 0 | 0 | 1 | 1 | 0 | 0 | 3 | 12 | 23 | 122 | 360 | 500 | 95 | 13 | 1 | 0 | 0 |
| 75.79 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 10 | 27 | 62 | 337 | 362 | 58 | 7 | 1 | 0 |
| 80.84 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 5 | 15 | 14 | 71 | 259 | 216 | 28 | 6 | 0 |
| 85.89 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 3 | 3 | 6 | 29 | 110 | 81 | 9 | 1 |
| 90.94 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 3 | 8 | 29 | 7 | 1 |
| 95. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 2 |

Table 3.3 – Cross table of ages of both partners for the households "Couple without children" for Namur (INS : 92094). The age of the male corresponds to the rows and the age of the female to the columns.

The goal is to minimise the difference between the simulated age differences distribution and the theoretical one :

$$Error = min \sum_{i,j} |Obs_{i,j} - Theo_{i,j}|$$

where $Obs_{i,j}$ and $Theo_{i,j}$ are respectively the number of observed and theoretical couples with $i^{th}$ men age and $j^{th}$ wife age.

However, it is always better if the fitness function is bounded by 0 and 1, which is not the case of our proposed objective function. To bound by 1, it is sufficient to divide by the maximum value it can take. The value that can not be exceeded is two times the number of household, which is the sum of each cross-table ($Obs$ and $Theo$). Indeed, if a couple is missing in a category, this means that there is an extra couple in another age category. Each error could be reflected twice. Unfortunately, depending

on the structure of the theoretical distribution, this maximum could be unreachable. However, the important part of the function is that the target (error of zero) is reachable and clearly, if the good combination is found, this error function will be 0. Finally, the fitness also requires that 1 means a perfect fit and 0 a far from reality one. Thus, the relevant fitness is defined as :

$$f(Obs) = 1 - \frac{min \sum_{i,j} |Obs_{i,j} - Theo_{i,j}|}{2 * \sum_{i,j} Obs_{i,j}}$$

If the candidate of the beginning of the iteration is reported as $x$ and the proposed neighbour $x'$, the probability to keep $x'$ over $x$ with a temperature $T$ is (this is the *Metropolis condition* (Hoos and Stützle, 2005)):

$$p_{accept}(T,x,x') = \begin{cases} 1 & \text{if } f(x') > f(x) \\ exp(\frac{-(f(x)-f(x'))}{T}) & \text{otherwise} \end{cases}$$

This implies that when $T$ is constant, only the loss in the fitness function of replacing $x$ by $x'$ will be considered in the exponential, generating more chances to keep the worst candidate if it is not so distant from the $x$. The temperature influences the chances to keep worst candidates. It can be constant or decreasing over the iterations, either continuously or by steps.

Designing the temperature function can be difficult (Cohn and Fielding, 1999) and sometimes a fixed temperature could perform better than a cooling process (Fielding, 2000). Moreover, a bad temperature parametrisation could result in performances poorer than the Simulated Annealing with the probability to keep a worst candidate fixed to 0. For this reason, preliminary tests were implemented with a probability to keep a worst candidate equals to 0.

The first step of the Simulated Annealing, the random generation of a combination of couples is simulated once and all tests of different neighbourhoods definitions and temperature start from this initial point. This avoids a huge bias of better random initial points. The first attempt defines a neighbourhood candidate as the same households configuration with just two couples exchanged. Figure 3.6 shows the resulting fitness that increases but takes already 80.000 iterations to reach a fitness still under 0.9. This really slow convergence suggests that the chosen neighbourhood could be better designed. We tested several number of couples exchanges and Figure 3.7 indicates that more exchanges accelerated the fitness increase.

However, only 2000 iterations are generated and it is reasonable to assume that with the process moving forward, exchanging too many couples per iteration can block the convergence. When iterating further, as observable on Figure 3.8, one exchange outperforms all other neighbourhood proposition.

Our convergence could perhaps be accelerated with a decreasing number of exchanges over the iterations. Indeed, the aim is to always follow the fitness with the

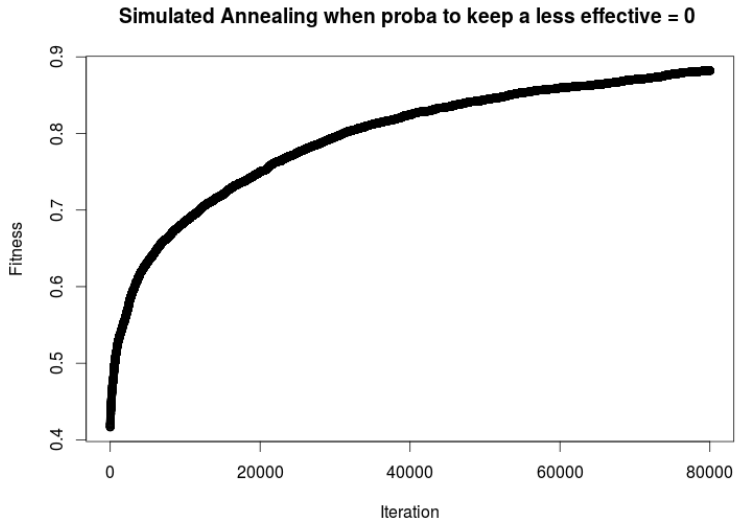


Figure 3.6 – Probability to keep a worse candidate = 0 - Neighbourhood = 1 couple exchanged

highest tangent slope. Zoom into the left part of the graph indicates that 20 exchanges directly behaves as 15 exchanges, meaning that 20 is already too high for the first stages of the algorithm. Several discrete function have been tested, and the better try involves a step function defined to always keep the number of exchanges implying the highest tangent slope (called $NE(i)$ for Number of Exchanges at iteration $i$):

$$NE(i) = \begin{cases} 15 & \text{if } 0 < i < 1000 \\ 10 & \text{if } 1000 \leq i < 1500 \\ 5 & \text{if } 1500 \leq i < 3000 \\ 3 & \text{if } 3000 \leq i < 4000 \\ 2 & \text{if } 4000 \leq i < 7000 \\ 1 & \text{if } 7000 \leq i \end{cases}$$

The resulting fitness, compared to the fixed neighbourhood in Figure 3.9, performs better at the early stage of the implementation, but the curve of one exchanges catches this curves, with both ending in a similar shape. Note that for Dumont et al. (2017*c*), only 6000 iterations were generated and we favoured the interval version of neighbourhood's definition. However, in the following, we consider one exchange since the final results are similar.

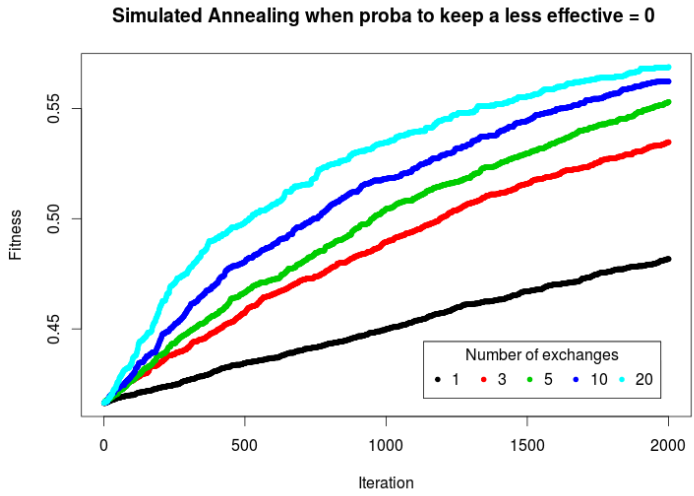


Figure 3.7 – Probability to keep a worse candidate = 0 - Neighbourhood = 1, 3, 5, 10 and 20 couples exchanged for 2000 iterations

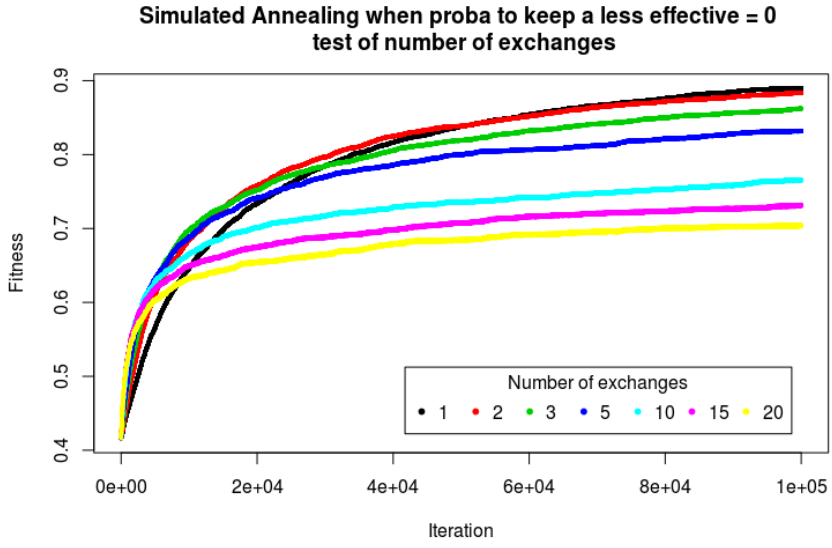


Figure 3.8 – Probability to keep a worse candidate = 0 - Neighbourhood = 1, 2, 3, 5, 10, 15 and 20 couples exchanged for 100.000 iterations

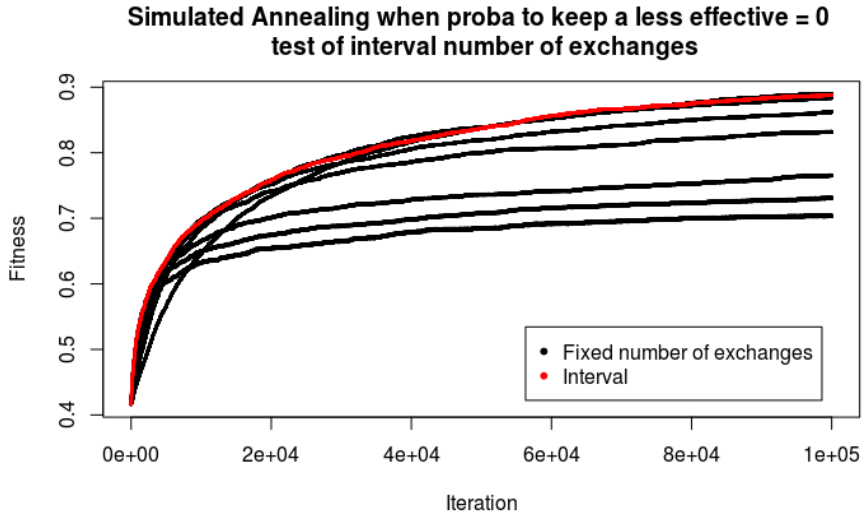


Figure 3.9 – Probability to keep a worse candidate = 0 - Neighbourhood = 1, 2, 3, 5, 10, 15, 20 and interval couples exchanged for 100.000 iterations

Having designed an efficient neighbourhood through the iterations, we focus next on the definition of the temperature to introduce the possibility to exit from a local optimum. However, the aim is to find a temperature ensuing a better fit than the algorithm keeping systematically the best candidate. A wide range of temperature functions have been tested. We present first a naive linear decreasing proposition (with $i$ the iteration)

$$T(i) = \frac{Max_{iter} - i + 1}{Max_{iter}}$$

Thus, with the iterations increasing, less chances are left to a worse candidate. The associated probability to keep a worst candidate ($exp(\frac{-(f(x)-f(x'))}{T})$) is illustrated on the left in Figure 3.10. The probability 0.5 corresponds to the white zone on the Figure, meaning that both candidates have the same probability to be saved. Moreover, the pink and blue zones mean respectively that more chances are left to the better/worse previous candidate. The blue zone being quite well extended, a coefficient is added on the denominator of the temperature function to reduce this area. We randomly tested 5 here, to test if it changes the results. Both temperature functions are generated over 50.000 iterations starting from the exactly same initial coupling (Figure 3.11). As comparison point, the best simulation without any chances to keep a worse candidate is included on the graph. No improvement appears with the addition of the possibility to keep a worse coupling associated to this linear temperature. We tested different other temperature functions and decided to keep a simple geometric version of the cooling function that has been efficient in different cases (Kirkpatrick et al., 1983).

The temperature update is thus

$$T := \alpha T$$

which require to calibrate the initial temperature $T_0$, the rate $\alpha$ and the number of iterations $n$ performed before updating the temperature. Table 3.4 indicates the parameters tested. Since all possible combinations of these parameters are implemented, this represents 75 different simulations.

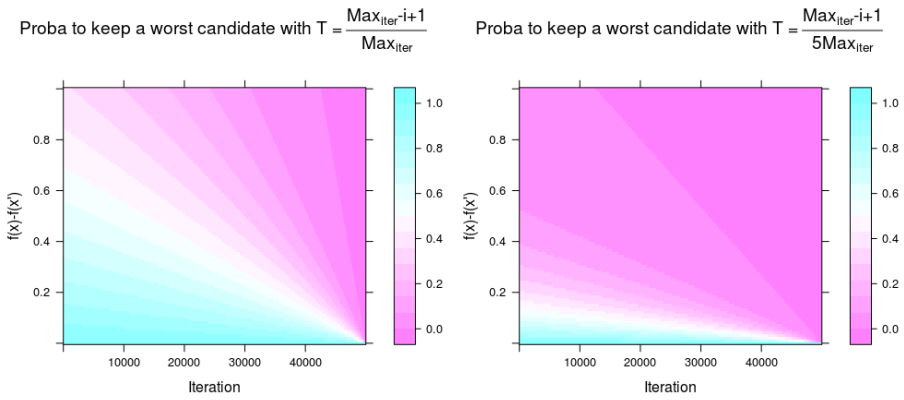


Figure 3.10 – Probability to keep a worse candidate - Naive linear temperature proposition

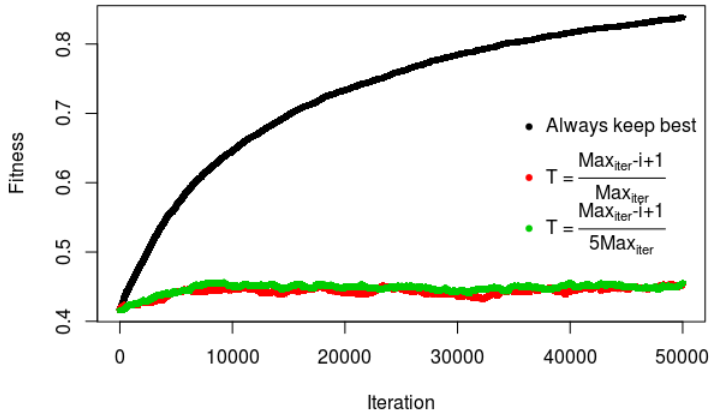


Figure 3.11 – Resulting fitness from Simulated Annealing with naive linear temperature function

| Parameter | Tested values |
|---|---|
| Number of iterations $n$ with same T | 1, 5, 10, 50, 100 |
| Rate $\alpha$ | 0.9, 0.95, 0.99 |
| Initial temperature $T_0$ | 0.1, 0.25, 0.5, 0.75, 1 |

Table 3.4 – Values of tested parameters for the Simulated Annealing with the geometric evolution of temperatures

To analyse the best set up, different visualisations have been tested and we observed a clear tendency when separating the graphs per $n$ and the colors per $\alpha$, as displayed on Figure 3.12. This illustration shows that the choice of the number of iterations with same temperature and the choice of the rate $\alpha$ influence together the results. First, $n$ increasing implies poorer fitnesses for all different rate $\alpha$. Then, for high $n$, the smallest $\alpha$ seems better, whereas for small $n$, a higher $\alpha$ performs better. Thanks to these observations, we choose to keep $n = 1$ et $\alpha = 0.99$.

For the determination of the initial temperature $T_0$, the curves concerning our choice of the other parameters are isolated and coloured depending on $T_0$ on Figure 3.13. No huge differences appear between the different initial temperatures. A very tiny tendency of better results for small $T_0$ induced the choice to keep $T_0 = 0.1$. The fitness never seems to exceed 0.9. This means that there are still couples in a "wrong" age differences category. The final parameters are relaunched over 250.000 iterations for 5 different runs to check the evolution of the fitness further and see if the different runs continues to behave similarly. Figure 3.14 confirms the stability over different runs, but shows also that the fitness doesn't increase rapidly to approach 1. The maximum fitnesses for the 5 runs are very close to each other (0.9348945, 0.9375965, 0.9318065, 0.9366958 and 0.9351518, respectively corresponding to 1012, 970, 1060 and 1008 couples in the wrong cross category of ages, out of 7772 couples to form), which could mean either a local maximum, or that just few remaining individuals could be exchanged to improve the fit and the random exchanges have less chance to find the good remaining exchanges. Nonetheless, this could also indicate that our definition of fitness never could reach a larger value than this one.

A possible explanation of the impossibility to reach 1 is that the age distribution between couples and the ages available in the pool of individuals are not exactly consistent as we can see on Table 3.5. Small differences appear, which influence the coupling. Indeed, we have an absolute total difference of 134 males in wrong age category, even if the total number of male corresponds. By checking the same information for the female, the same statements of 134 wrong classified females is observed. This imply a potential of 268 couples impossible to include in the right cross category of age, leading to a maximum reachable fitness of 0.9827586. This value is largely higher than the final fitness of our Simulated Annealing simulations.

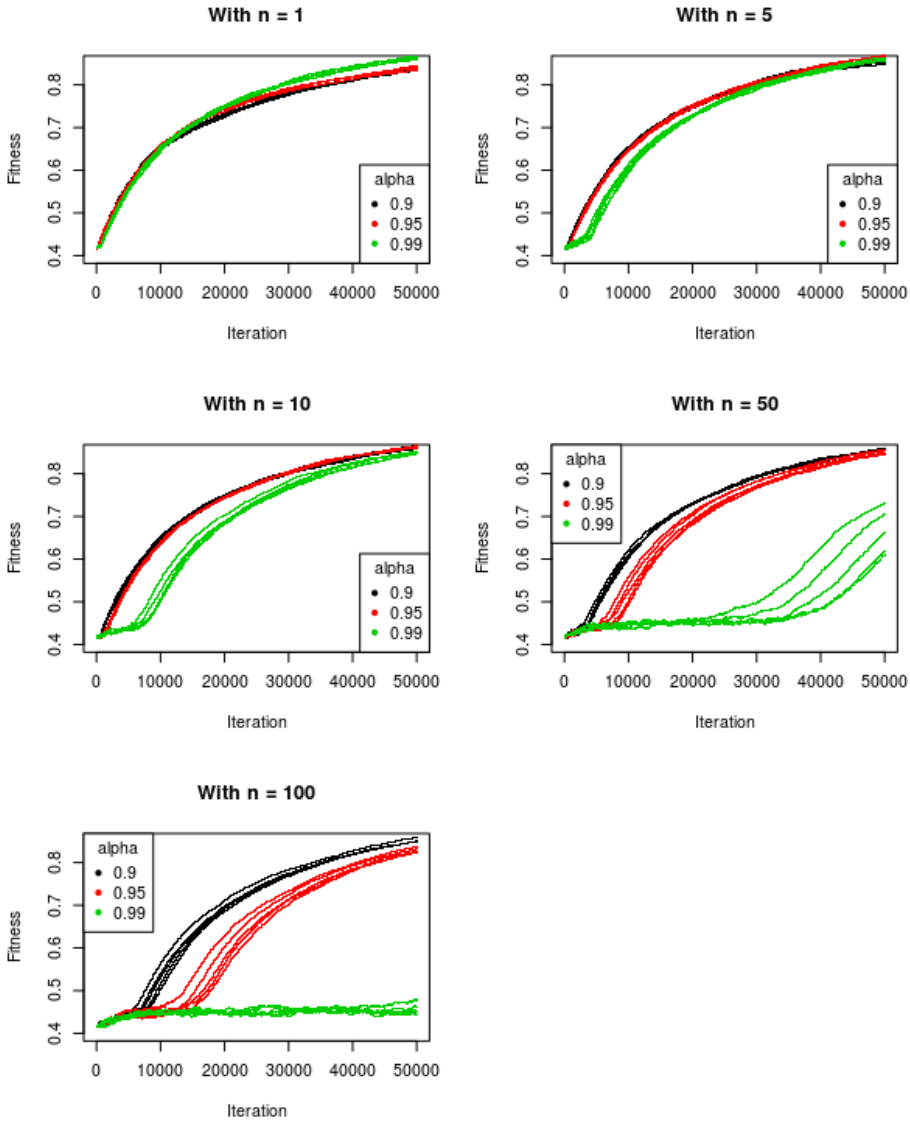


Figure 3.12 – Fitness resulting from Simulated Annealing with geometric temperature - test of parameters (color = $n$)

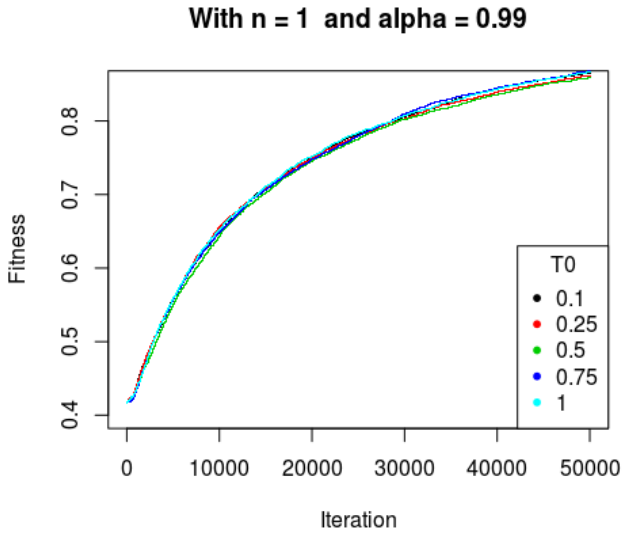


Figure 3.13 – Fitness resulting from Simulated Annealing with geometric temperature with $\alpha = 0.99$ and $n = 1$ (color = $T_0$)

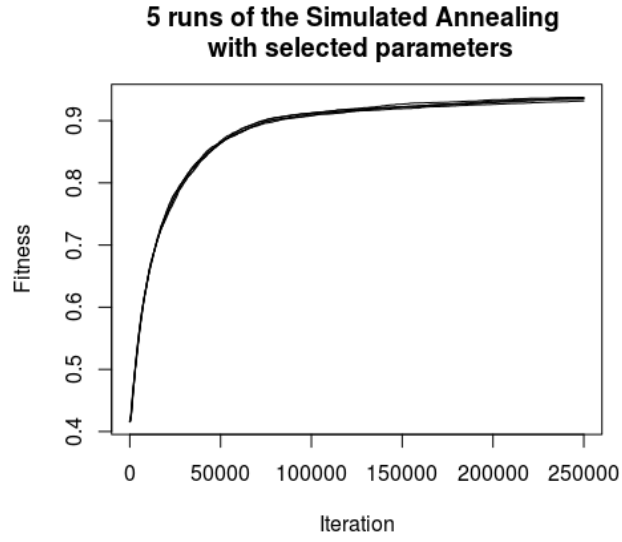


Figure 3.14 – Fitness resulting from Simulated Annealing with geometric temperature with $\alpha = 0.99$, $n = 1$ and $T_0 = 0.1$ (5 runs over 250.000 iterations).

| Age category | Counts in pool of individuals | Counts in age differences |
|---|---|---|
| 15.19 | 0 | 3 |
| 20.24 | 31 | 43 |
| 25.29 | 229 | 235 |
| 30.34 | 193 | 185 |
| 35.39 | 136 | 124 |
| 40.44 | 127 | 119 |
| 45.49 | 171 | 172 |
| 50.54 | 404 | 426 |
| 55.59 | 935 | 957 |
| 60.64 | 1485 | 1484 |
| 65.69 | 1122 | 1111 |
| 70.74 | 1145 | 1131 |
| 75.79 | 870 | 868 |
| 80.84 | 625 | 615 |
| 85.89 | 242 | 243 |
| 90.94 | 52 | 51 |
| 95. | 5 | 5 |

Table 3.5 – Distributions of age category for male in type of household 'Couple without children' for Namur (INS : 92094)

An important remark is that, for the sake of computational facilities in testing more different parameters, each configuration (proposed in Table 3.4) has been tested only twice to ensure no huge differences caused by the random exchanges (but the illustrations contain only one run for each calibration set). Run this code 20 times to perform averages and standard deviations could give more robust results, but in this study, it has been decided that the calibration is too much time consuming.

This method would require a more deep investigation and could be easily adapted to multi-objective problems. This type of heuristic methods never "stops" since we can not be sure to have found the best calibration and to have reached the global optimum of the function. Since the original Walloon project needs rapidly a static synthetic population to allow considering the time evolution, no other tests have been undertaken for this part. We already generated a reasonably good coupling, but we would like to see if a simpler method couldn't give similar or better results. Moreover, the Simulated Annealing has been constructed only for the couples without children in Namur, whereas a total Belgian synthetic population is necessary. Note that the fitness associated to the random draw performed in next section is 0.9809573 (for Namur).

#### 3.2.2.2 Random draws in the distribution

The Simulated Annealing being quite difficult to set up and parametrise, a more basic and simple probabilistic method is proposed in this section, involving other issues discussed at the end of the section. Note that the Simulated Annealing was performed only for couples without children, whereas the probabilistic association discussed in this section is performed for all types of household.

As for the Simulated Annealing, each head is associated to an household and, for all type of households, except the isolated ones, the rest of the household members need to be completed. The challenge is to satisfy the size, type and role in the household constraints, as well as the age distributions in the couples and between head and children.

**Households without children**

For the couples without children, we have a database containing, per municipality, the information about the number of individual per role in the household and the gender of head and partner. The concept of the method in this case is schematised for the couples with a male as head of the household in Figure 3.15. This procedure is totally similar for the couples with a female at the head of the household.

First, thanks to the role in the household, two groups are formed: the male heads and the female partners. Then, the process starts to iterate. The age distribution of the remaining available partners is generated to check if all age categories are still available. Simultaneously, a still available head is randomly picked. Depending on his age, the age distribution of his potential wife is generated, by keeping only the ages containing at least one still available woman. If there is still at least one free woman in the probable age for the head, an age for a wife in randomly picked inside the distribution. Then, a corresponding woman is assigned to the household. Otherwise, if no probable woman is available, this head stays alone and the process restarts randomly choosing another head. The final step of each iteration is to adapt both pools of individuals. The loop ends when a pool is empty.

For the cohabitants without children, no additional statistics giving information on how to group the individuals are available. For this reason, only the role in the household is used and a cohabitant is randomly associated to each head.

**Households with fixed number of children**

The households taken into consideration now are the couples with children and the monoparental families. The creation of the couples inside these households are initially performed exactly in the same way as for the couples without children. The cohabitants with children are discussed separately since with the attributes available, we are not sure about the number of adults and children to include in each household.

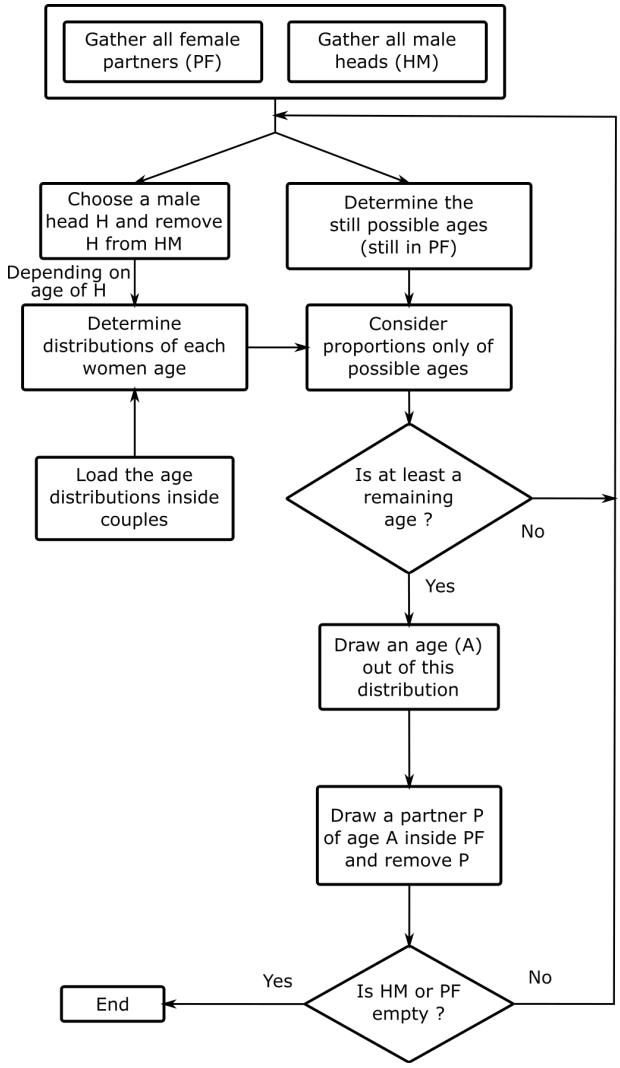


Figure 3.15 – Diagram of the random probabilistic determination of couples (case with male heads)

For the couples with children and the monoparental families, the number of children to assign to each head can be calculated, knowing the size of the household. To associate the children to the households, we have the cross table with the municipality, type of household, age of the head and age of the child. The assignment of the children follows the same logic as the one of the partners, except that it generates the desired number of children instead of just one. In a first stage of the generation, the algorithm

forces the households to include only members needing the same number of cohabitants as the head. However, constrain the algorithm with this condition creates too few households and several individuals stay unassigned. A second stage is then executed with the remaining individuals without checking the desired size of household, but only focusing on the age differences between the head and his/her children.

**Cohabitants with children**

The cohabitants with children are particularly different to generate, because we can not directly determine the type of individuals needed for each head. For example, if a head needs three cohabitants, it could be an adult and two children, or two adults and one child. The structure of the household is thus unknown. The only certitude is that it is not a couple with children neither a head with only children. Using this fact, we first ensure that each head is associated to at least one child and an adult different from his/her wife/husband. The only table helpful here is the age difference between the head and the children. The association of the children is performed similarly to previous section and the second adult is chosen randomly. Then, the households are completed randomly with the remainder persons. As for the couples with children, we first generated the households strictly constrained by its size, before to complete the households without considering the size of household, still trying to respect the age distribution between head and children.

**Collective and other households**

The individuals in collective and other type of households are not grouped into households, because first, we don't have access to satisfactory data about the size of households (a lot being of size "6 or more") and, more important, their are not the target population of the research asked for the Walloon Region. Indeed, their aim is to analyse the health needs of elderlies in the future years to adapt the offer of health services such as retirement houses or home assistance. We then suppose that the persons included in a collective households are already assisted. Note that the individuals in collective or other households only represent 1.075% of the Belgian population.

**Quality of the generation of the households**

The algorithms described in this section are generated for each municipality allowing us to analyse the results for each target feature of the grouping. Note that this process is probabilistic with a lot of random draws and needs to be generated several times if we want to make sure to not reach a population that is not optimal because of many unlucky draws. The execution is thus repeated 4 times to avoid this. However, there is no need to test deeper the stability and the robustness, since the aim here is to reach one acceptable population in a reasonable time, and we are not estimating unknown features (which would need several runs to be estimated by their mean) but trying to simulate one population respecting some constraints. First, Figures 3.16 and 3.17 show the simulated and desired counts for both age difference tables (respectively

between partners and between head and child) for all municipalities and all type of households. By checking the vector of differences, we confirmed that the fit for these two graphs is perfect. The four different runs resulting in exactly the same fit (in all categories, the simulated counts equals the theoretical count) for both age tables, the graphs contains only one simulation.

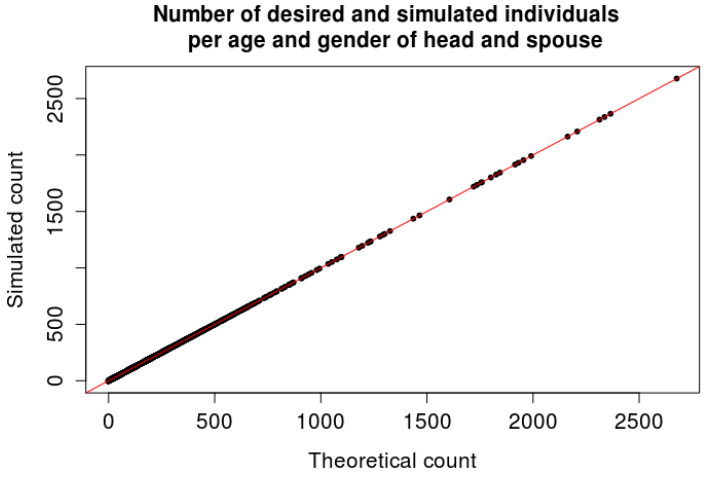


Figure 3.16 – Accuracy of the probabilistic households in term of age between head and partners for married couples

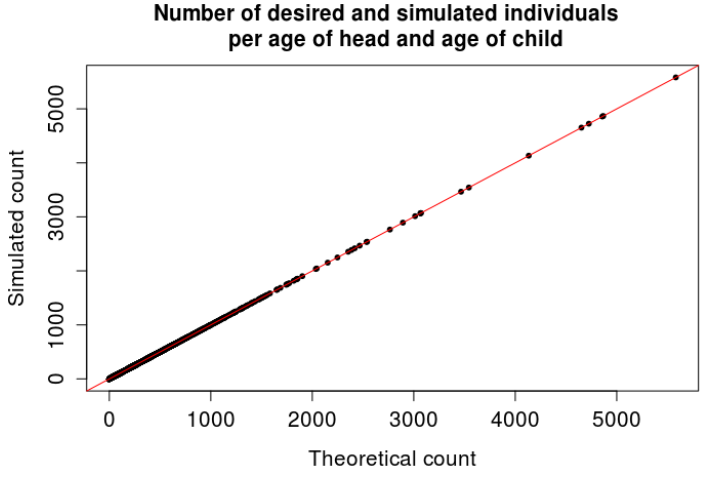


Figure 3.17 – Accuracy of the probabilistic households in term of age between head and her children

However, we decided that respecting these age distributions was more important than respecting the sizes of households. It is thus pertinent to quantify the number of households with individuals assigned to the wrong number of cohabitants. Note that the size category '6 and more' is not taken into account here, since the precise size is unknown in this case.

For this indicator of the number of individuals in a wrong size of household, the different runs give slightly different results. Few municipalities are included in the category of more than 1000 persons assigned to the wrong size of household. However, in Belgium, the counts of individuals per municipality is highly heterogeneous. For this reason, the maps in Figure 3.18 illustrate for the 4 different runs and for each municipality the proportion of individuals in a household of the wrong size. All municipalities include a very low proportion of individuals assigned to the wrong size of household, with a maximum of less than 3% for a tiny municipality of only 1878 inhabitants. This small error is acceptable, since we are aware that the two different sources of data are not exactly coherent with each other, and as discussed at the end of the section on Simulated Annealing, a perfect population respecting simultaneously all data is impossible.

To allow an easier comparison of the 4 different runs, the used intervals are the same on each map. We observe better results in Flanders and Brussels than in Wallonia for all runs. This can be explained by the differences of density of the population, since more individuals intuitively implies less impact induced by few wrong sized households. The vast majority of municipalities are in the lower classes for all runs. However, some municipalities, such as Doische or Ohey, are in the 4 differents simulations included in the class of between 2 and 2.5% of individuals in a wrong sized household. For the province of Luxembourg, the first two runs result in a higher proportion than the two last runs. Note that the process is computed independently for each municipality and the final need is to have one satisfactory static synthetic population. Thus, a different run can be chosen for each municipality.

Note that in addition to the individuals in collective or other types of households, some individuals stay not assigned in the categories "Cohabitants with children" and "Cohabitants without children". This is caused by small inconsistencies in the data, since, for example, we have an odd number of persons to include in households of size 2. The number of individuals in this case (without considering the collective and other types of households) is generated per municipality for the 4 different runs and reported in Table 3.6. For all runs, exactly 328 municipalities (or 55.7% of the municipalities) do not contains individuals not assigned to a household. Very small differences appear between the runs indicating that the inconsistencies in the data is not responsible for all not assigned individuals. The associated proportion of individuals in this case per municipality stays very low with the highest rates observed for Baarle-Hertog and Martelange, with proportions (for all runs) of 0.41% and 0.26% respectively.

In conclusion, the different runs give similar results and we chose to continue with

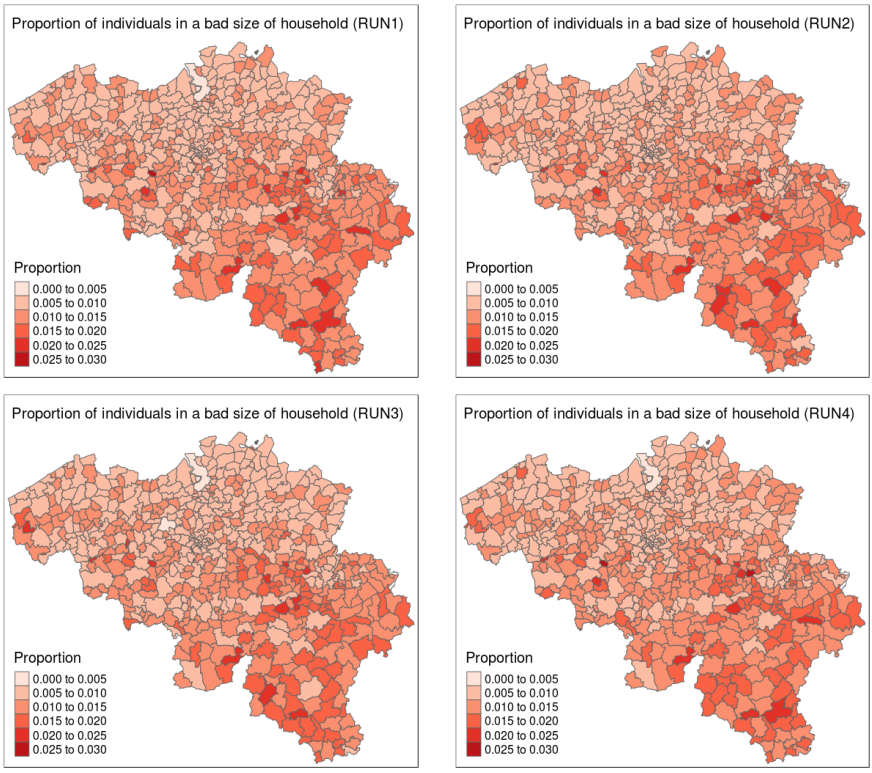


Figure 3.18 – Proportion of individuals assigned to a household of the wrong size

| | 0 | 1 | 2 | 3 | 4 | 6 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| Run 1 | 328 | 15 | 142 | 8 | 65 | 15 | 10 | 5 | 1 |
| Run 2 | 328 | 15 | 142 | 8 | 65 | 15 | 10 | 5 | 1 |
| Run 3 | 328 | 15 | 142 | 8 | 65 | 15 | 10 | 4 | 2 |
| Run 4 | 328 | 15 | 141 | 9 | 65 | 15 | 10 | 5 | 1 |

Table 3.6 – Counts of municipalities depending on the number of individuals not assigned to a household

the first run for the dynamical evolution. Mixing the runs to keep the best one for each municipality could improve slightly the synthetic population, but has not been performed for this project.

## 3.3 Time evolution

Let me remark that this part has been developed in collaboration with William Henrotin who was engaged on the Walloon Region contract and worked on this project for one year. The project is coded in C++ and run on the Hercules cluster of CECI.

### 3.3.1 General process

Having a complete initial population, the next step is to make these agents evolve through time and space. For this study, rates for all considered life events are available per year and the project aims at forecasting yearly the population until 2030 (19 simulated years). For this reason this framework is implemented as a discrete time model with a timestep of one year. The global evolution is split into different sub-modules corresponding to key events chosen specifically for the purposes of this study. We have individual processes : each year, each agent ages and some agents die. The other considered processes concern the households : a household can welcome a new baby, a marriage could merge two households, a divorce separates an household in two, the complete household could move inside or outside Belgium and new households could arrive from abroad.

An important remark is that the individuals in the starting synthetic population are characterised also with their professional status and highest educational level. However, for the former, a yearly upgrade seems not to be appropriate and no rates are available to estimate an update of this attribute. Moreover, the Walloon Region is interested in the health needs of the elderlies, which are almost all professionally inactive. Thus, we decided to forget this variable. For the educational level, the assumption is made that the target population in this study (the more than 60 years old in 2030) will not reach a new educational level between 2011 and 2030, since they are already 40 years old at the beginning of the simulation (in 2011). Thus, the attribute is kept, but with a high attention that the variable is now "highest diploma level in 2011".

The model considers six modules: ageing, death, birth, marriage, divorce and migrations. The whole population pass through a module before activating the new module, because modules such as marriages involve several individuals and several households. The order of application of the modules need to be then specified. For VBIH, the executed order is : Deaths, Ageing, Births, Migrations, Marriages and Divorces.

Performing the ageing before the births is necessary since the initial population contains babies of 0 year old. If we execute first the births, the babies of the first simulated year are added to the population and we have an artificial peak of babies. Moreover, deaths are performed before births, because we have for the births the rates corresponding to the babies still in the population at the end of the year. Then, divorce in the year of the wedding seems more probable than get married the year of a divorce. The rest of the order is a decision taken just intuitively. Note that the influence of this

choice is analysed in Chapter 4.

The global process is illustrated in Figure 3.19. It highlights the order of the modules and (in blue) the summary of the input needed for each of them. Each module is detailed further in this section.

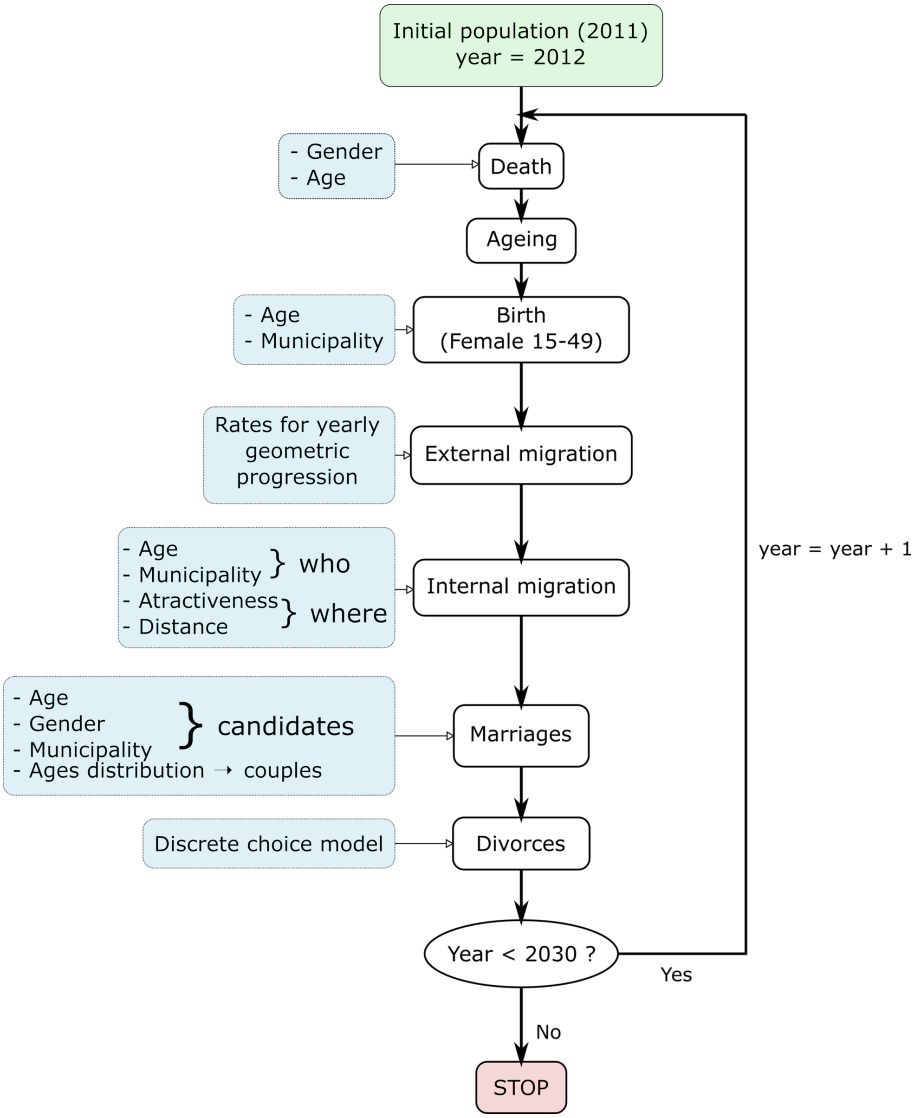


Figure 3.19 – Global process of time (and space) evolution for VBIH

Ageing is simple with a yearly timestep : each agent is one more year old every year. For the other modules, a first naive attempt consists in considering each process as probabilistic, with fixed rates estimated for 2011. Fixed probabilities would be naive since the rates[5], such as for example the mortality and the fecundity, are evolving through time. Each module is developed in a following section. Note that the global assumptions are mimicking the global assumptions of the macrosimulation performed by the "Bureau Fédéral du Plan" (Duyck et al., 2014).

### 3.3.2 Death

The evolution of the death rates are estimated thanks to a negative exponential over time calibrated per age from 1991 to 2012 and extrapolated until 2030 (Duyck et al., 2014). Thanks to these rates, each agent has a probability to die during the simulated year (depending on her gender and age). Thanks to a random number generator with a uniform density between 0 and 1, the algorithm "decides" for each agent if she will survive the simulated year. The dying agent is directly removed.

### 3.3.3 Birth

Then, for the birth, each synthetic woman in age to procreate (15-49 years old) has a probability to give birth depending on her municipality and age. Note that the birth rates are not aggregated at the national level, since clusters of fertility have been created thanks to socio-economic characteristics (Costa et al., 2011). A rate adaptation similar to the same ones of the "Bureau Fédéral du Plan" (Duyck et al., 2014), namely the fact that the fecundity rates (per woman age and municipality) are decreasing from 2011 to 2015, but should begin to re-increase after to stabilise around 2020. To calculate the adapted rates, a correction coefficient is generated for each year to reach the resulting rates of the macrosimulation performed by the "Bureau Fédéral du Plan". Note that for the other modules, we rely only on the same assumptions as them, while here we rely also on their results. The used adaptation coefficients are guessed by observing the assumptions of the BFP. We made this choice to deliver the framework software in the due time, but this is of course not the best way to proceed and could be improved in a forthcoming work. When a woman has a baby, he is added to her household. This baby has a fixed probability of being a boy (of 0.5119) defined by the ratio given by the group DEMO (based on the national register for 2011). All needed characteristics are adapted (the size, type of household, etc).

### 3.3.4 Migrations

We have to consider inside and outside migrations. Indeed, the inside migrations imply to welcome the household in the arrival municipality, whereas the outside migrations induce the creation or removal of an household. First, for outside migrations, the model is basic, using a probabilistic process based on the number of outgoing

[5] Statistics of the SPF Economie, `http://statbel.fgov.be/fr/statistiques/chiffres/` and the Census of 2001

and incoming individuals per municipalities and per age. These rates evolve during the simulation with the outgoing population staying proportional to the growth of the population and the incoming population following an annual geometric progression starting with the known incoming population for 2011, controllable with the rate of the geometric progression. These upgraded rates are used to determine the number of individuals per age that should arrive from outside Belgium to this municipality and that move outside Belgium. Having these number defined, households are randomly chosen and if someone in the household has an age that should lead her leave the municipality, the total household is deleted and the number of individuals to remove is updated. For the incoming households, we have no information about the structure of the household, so we clone an existing household and perform the same way as for the leaving ones.

To simulate the internal Belgian migrations, a database contains for each possible combination of municipalities the number of moves per age, for the years from 2008 to 2013. This table has been used to generate two other tables : the attractiveness of a municipality (the average yearly number of individuals arriving in this municipality after an internal move) and the distribution of the distances (in terms of number of adjacent municipalities) between the departure and final municipality. Thus, with rates per municipality and age, that we divide by the number of members in the household (since the probabilities are available per individuals, but if someone moves, her household follows), we determine the households leaving the municipality. Then we need to choose a destination. For this, we first determine the distance that the household will travel (with the distance database) and then, weighted by the attractiveness of each municipality in this distance, we chose the destination.

### 3.3.5 Marriages

The economic theory around the analysis of the marriages is called the "Marriage Market" (Chiappori, 2020). The marriages are here simulated in two steps. The first one isolates all potential newly weds in the year under scrutiny. The second one forms the couples from the pool of candidates. In the first stage, the identification of marriage candidates is achieved thanks to rates per age group, gender and municipality extracted from data of the National Register (delivered by the group DEMO-UCL). These rates are related to all events changing the civil status of a person. Here we are interested in modelling the marriages, meaning that only the changes "from singles to married" and "from widowers to married" are relevant. Each potential candidate (the singles or widowers individuals) has thus a probability to get married depending on his/her age, gender and municipality and depending on a random draw in this distribution, he/she will be added or not to the pool of potential newly weds. Note that we assume children under 18 don't get married. Their marriage rates are thus equal to 0. Once we have gathered the two subgroups of men and women candidates for marriages, the next step is to create the couples. One after the other, every man in the pool is considered. For each selected man we consider his associated wife's age distribution and pick an age difference randomly from this distribution. Then a woman so aged is

selected randomly to form a couple with the man. Note that only the age difference is considered, but not if it should be the wife or the husband who is the oldest. The process stops when either the women pool is empty or all men have been taken into consideration. For each marriage, a new household is created with the two partners, and the children which are possibly linked with one of them. Their old household is updated after their leaving.

### 3.3.6 Divorces

The divorces in VBIH have been generated thanks to a logit discrete choice model, using the notion of utility (=satisfaction) for each possible choice, depending on dependent observed variables and on a random part simulating the unobserved preferences ($\varepsilon$). Only the differences among the utility values do matter in the model. Indeed, add a fixed number to all utilities will not change the results. For this reason, a possible choice is chosen as reference level, and its utility is fixed to 0. This allows all other alternative to be determined on this basis and avoid that the estimators of the model results in high standard errors. The Logit modelling performed here considers the random terms $\varepsilon$ as independent and identically distributed following a standard Gumbel law. This specification ensure a simple form of the probability of each alternative (Ben-Akiva and Lerman (1985)), expressed here for the specific binary case and individual $i$:

$$P_i^{div} = \frac{e^{V_i^{div}}}{1 + e^{V_i^{div}}}$$

Once we have the calibrated utilities, this probability is used to determine for each couple if they divorce during the simulated year or not. To simulate the divorces, we have a dataset with these informations about each couple:

- wife and husband's age categories,
- diploma level of both members,
- their household's size,
- the actual duration of the wedding.

Fixing the utility to stay together to 0, we estimated the utilities (using Biogeme (Bierlaire, 2003)) with all available variables and removed the non significant ones (in terms of the statistical t-test). The final used utilities are respectively for the choices "divorce" or "stay married" :

$$\begin{aligned} V^{div} &= -1.5W_3 - 1.75W_4 - 1.79W_5 - 1.86W_6 - 2.25W_7 - 2.88W_8 \\ &\quad -3.52W_9 - 4.1W_{10} - 4.54W_{11} - 5.07W_{12} - 6.85W_{13} \\ &\quad -0.23M_{age} - 0.15M_{dipl} - 0.16HH_{size} \\ V^{stay} &= 0 \end{aligned}$$

With $W_k$ being a boolean equal to 1 when the woman is of age category $k$ (with the third age class being 15-19 years old, fourth 20-24, etc.); $M_{age}$ being the age category

of the man, $M_{dipl}$ the highest education level of the man and $HH_{size}$ the size of the household. This is the resulting discrete choice model, when removing the insignificant variables (woman age category higher than "70-74" years old, the duration of the wedding and the diploma of the woman). Remark that the influence of an additional category in female's age is not linear, whereas it is for the male. This is the better result of several tests with all possible combination of boolean variables.

Note that this part of the Walloon Region project raises several research questions that has been investigated after the end of this contract. For this reason, we later analysed more precisely the case of the divorces modelled with a discrete choice model and a feedforward neural network. Further information and a more detailed explanation of discrete choice models are thus left for Chapter 5.

## 3.4 Quality of the results

### 3.4.1 Validation

The macrosimulation performed by the "Bureau Fédéral du Plan" (Duyck et al., 2014) considers a lot of different processes and is a reference in terms of demographic simulation of the population. For this reason, comparisons between our model (aggregated for the comparison) and their macrosimulation could be a suitable basis to validate our model. If the latter agrees with their simulations, this means that we forecast the same aggregated population, but having in addition a disaggregated version of this evolving population.

The results presented in this section are based on one simulation from 2011 until 2030. With the high number of events (deaths, births etc) in the microsimulation, the stochastic patterns of the probabilities processes would be balanced, since asymptotically, the observed proportions tends to the probabilities. However, a short analysis of the stability of the process over 10 runs is performed in the next section.

We verified that the starting population for 2011 of VBIH matches the population of the BFP for 2011. This is the case as illustrated for the individual ages distribution in Figure 3.20.

An important point of our framework is that it needs an assumption on the evolution of the external immigration. The basis year is 2011 and a coefficient is used to estimate the number of immigrating individuals every year. We first tested to perform the analysis without any foreigners entering Belgium (left panel of Figure 3.21) and realise that without any immigration, the growth of the Belgian population would remain almost constant. However, this is not what the forecasting of the BFP reports. We thus decided to include the immigration and different coefficients have been tested. Finally, we decided to keep the evolution of immigration corresponding always to 99% of the immigration of the previous year.

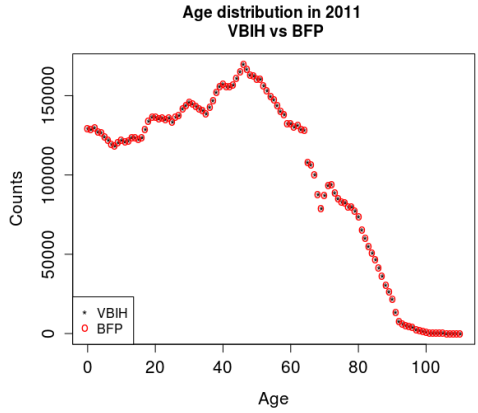


Figure 3.20 – Age distribution in 2011 : comparison of VBIH and BFP

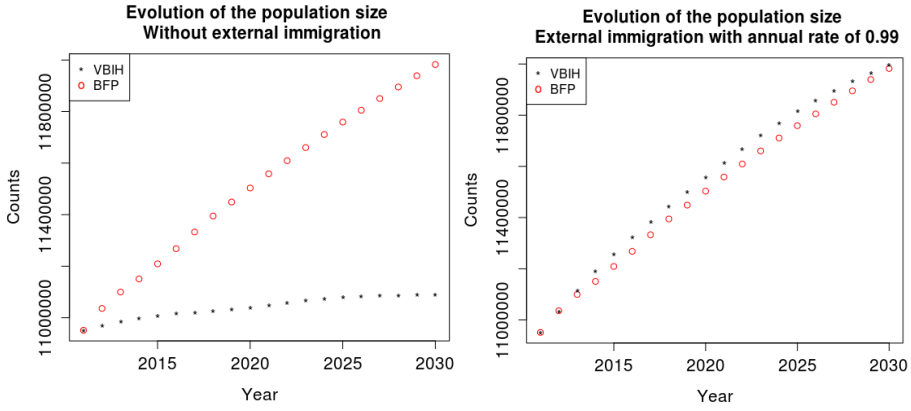


Figure 3.21 – Evolution of the total population size : comparison of VBIH and BFP depending on the rate of external immigration

Note that this can be explained by the fact that the international immigration was at a very high level in 2011 with the political context of several countries at this moment (Duyck et al., 2014). The right panel of Figure 3.21 illustrates the comparison between the total size of the associated population, better fitting the BFP forecast.

Another aggregated information we have from the macrosimulation of the BFP is the age distribution per year. Figure 3.22 includes the age distributions for 2020 and 2030. The trend seems similar for VBIH and BFP, but we have a bit more of individuals of 40 and more years old, whereas the BFP has a bit more 20-40 years old

individuals. Our birth process generates a bit less babies than the macrosimulation. This difference becomes bigger at the end of the simulation in 2030. Let us observe that at this point, we already have 19 simulated years and we are aware that this is a long term simulation for the assumptions done and the chosen calibration. We can thus conclude that these differences aren't alarming.

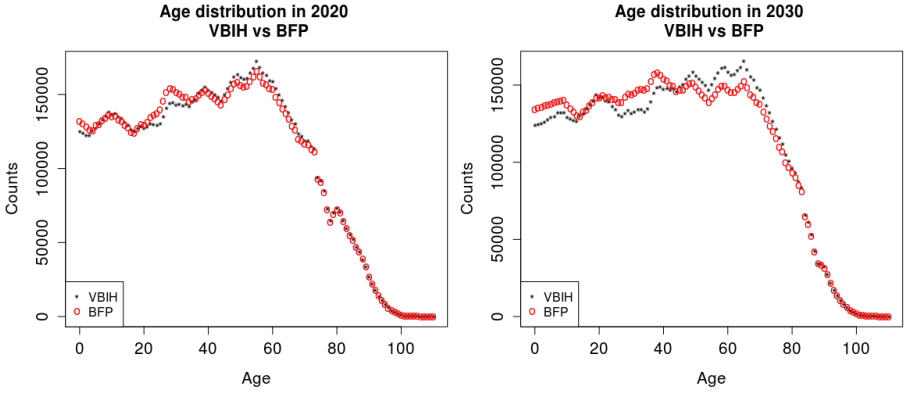


Figure 3.22 – Age distribution in 2020 and 2030 : comparison of VBIH and BFP

To ensure that these differences are not biased particularly for one gender, the age distributions per gender are shown in Figure 3.23. The differences between VBIH and BFP do not impact any gender in a particular way.

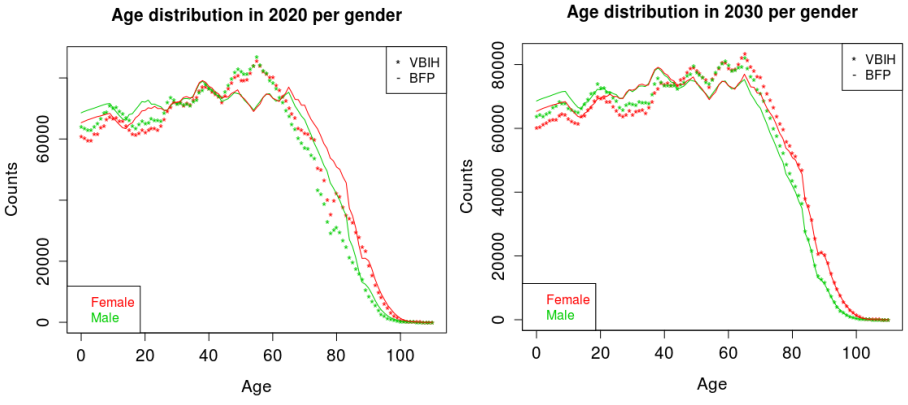


Figure 3.23 – Age distribution in 2020 and 2030 per gender : comparison of VBIH and BFP

This brief comparison is encouraging in terms of the quality of our simulations. We keep in mind that the framework could be improved with, for example, better adapted birth and deaths rates, but the results are already coherent and thus satisfactory for our objectives.

### 3.4.2 Stability

Before delivering the framework and beginning the analysis of the evolution of the population, we want to study if our model is stable amongst different runs. In particular, we check the influence of the pseudo random numbers generator' seed. To highlight the sensitivity to randomness, we performed the simulation using 10 different seeds and look at the population size at the date 2025. The average, minimum and maximum number of male and female per year over the 10 runs are illustrated in Figure 3.24. Note that to have a totally complete view, we added the associated simulation of the BFP.

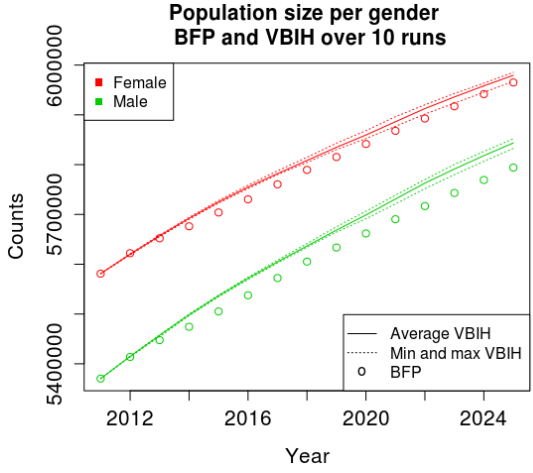


Figure 3.24 – Population size per year and gender: average, minimum and maximum for 10 runs of VBIH and BFP until 2025

For both genders, the simulations of the 10 runs stay closer one from the other, indicating no real impact of the seed of the random number generator. For the females, the VBIH simulations are closer to the one of the BFP than for the males. This could be investigated in more details if we want to better follow the BFP projections.

To confirm the stability, and not only at the aggregated national level, the standard deviation of the number of inhabitants of each municipality (over the 10 runs) is computed. The median is 49.69, meaning that half of the municipalities has a standard deviation of the size of its population below this value. The vast majority of the municipalities (84.9%) has a standard deviation below 100. Anderlecht has the highest

standard deviation with a value of 6.712. However, this indicator of the dispersion is dependent on the range of the variable and Anderlecht is a highly populated municipality. To have a relative indicator of the variation, the ratio between the standard deviation and the average of the final size of the population (thus without unit and not scale sensitive), namely the coefficient of variation, is available on Figure 3.25 (in a log10 scale). This figure indicates lower coefficient of variation for the municipalilites in Flanders than in Wallonia. Note that the highest coefficient of variation (12.21%) concerns Evere and all other municipalities are below 7.2% of coefficient of variation. This is high, but it is normal to observe more variations over the runs when performing a more disaggregated analysis. The analysis and interpretations of the final population per municipality for the simulated years thus need to be achieved with caution.

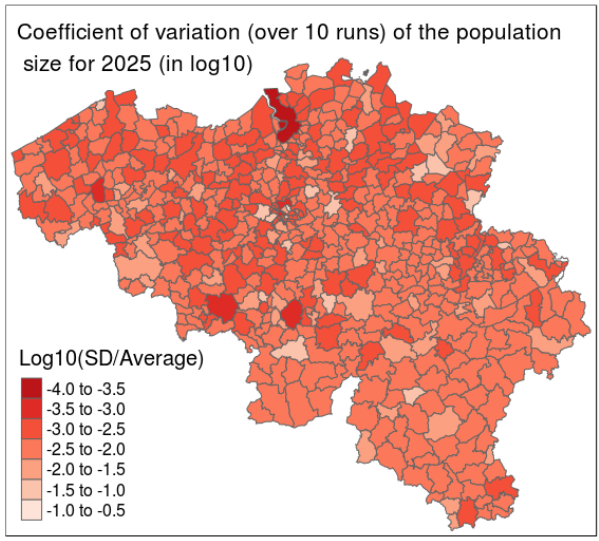


Figure 3.25 – Ratio of the standard deviation (over 10 runs) and the average of the population size for 2025

Determining if this variation is too high or not is a challenging task. When performing a chi-square hypothesis test on the homogeneity of the population size per municipality over the different runs implies the rejection of the null hypothesis for 2025. Executing the same test for each simulated year results in the rejection of the null hypothesis from 2017 to 2025. However, these kinds of tests are sensitive to the scale and considering the population size in thousand of individuals or in individuals changes the results. Analysing the horizon of good predictions is a complex task needing a complete analysis and the choice of a criteria indicating that the differences between the runs is now too large. This topic is not fully developed here, but Figure 3.26 contains already the boxplots of the coefficient of variation per year (in log10 scale). The direct observation is that the coefficients of variation do not explode.

It would be interesting to perform more runs on larger time horizon to check if this stabilises.

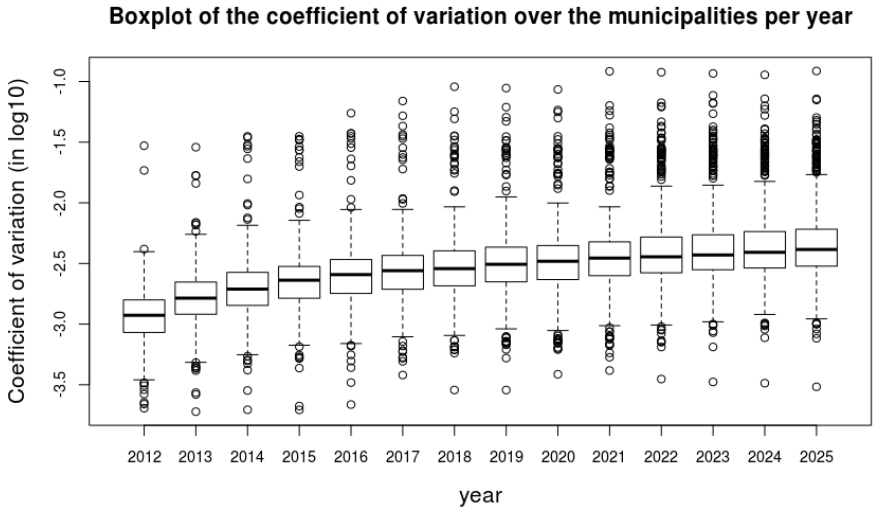


Figure 3.26 – Evolution of the repartition of the coefficients of variation (calculated over the runs for each municipality)

In conclusion, the aggregated counts are similar over the different runs, but more dispersion appears when considering each municipality alone. The user of the simulations needs to be aware of the potential differences between runs and the determination of a reasonable time horizon could be performed depending on different criteria such as the goal of the study and the precision needed.

## 3.5 Health data application

For the purpose of this project, we also determine other potentially interesting attributes that we could add to our population.[6] In order to do this, we choose several determinant illnesses highly impacting the elderly. They are grouped in bigger and broad classes and we have for example "chronic diseases", "Parkinson", "diabetes", etc. Thanks to data available on Pharmanet[7], we know the drugs, that have been reimbursed by the mandatory health insurance, as well as some characteristics of the people who bought these medicines. Thanks to the fact that some drugs are only used for one specific disease, these databases enable us to link some specific diseases

[6] This has been discussed with the demographers of the group DEMO of UCL and the health specialists of the OWS ("Observatoire Wallon de la Santé").

[7] "Statistiques sur les médicaments délivrés en pharmacies publiques" - INAMI

to some patterns of the population. These new datasets are used at the end of the simulation to estimate the proportion of diabetic person per municipality per year for example.

At the end of the project we provided the Walloon Region with an interactive framework software (with a simple interface) on which they can upload data of new health indicators per gender, age and municipality. The framework software then uses the simulated population to estimate the proportion (or counts) of persons concerned by the new indicator. It outputs a map and a database with the estimated numbers and proportions. For example, using data about the number of diabetic person per municipality, age and gender for 2010, the estimation is available for each future year. Figure 3.27 illustrates the results for the year 2011 and 2025. We can see that, under the implicit assumptions done here, the proportion of diabetic persons will increase. Of course, reasoning this way implicitly supposes that the proportion of diabetic person per age, gender and municipality will stay stable and that only the structure of the population in terms of these variables intervenes.

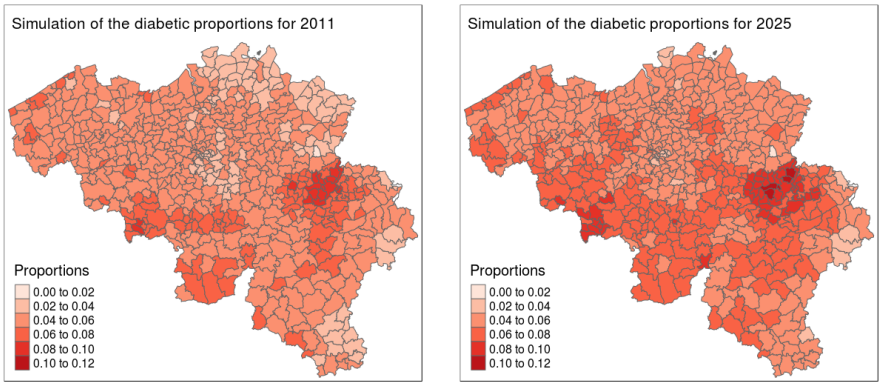


Figure 3.27 – Example of use of the platform for estimating the counts of diabetic persons for 2011 and 2025

Another disease interesting for the Walloon Region for which they can have data is the "bronchopneumopathie chronique obstructive (BPCO)". Again, considering the simulated population and the rates of affected persons per age, gender and municipality, an estimate, under the implicit assumptions that the disease prevalence is determined by the explanatory variables in use, is performed and is shown on Figure 3.28. Some municipalities seems to be more affected in 2025, but the differences aren't so large than for the diabetes. This depends on the correlation between the prevalence and the considered explaining variables (age, gender and municipality here).

The global analysis of the health results and the choice of new related data is left

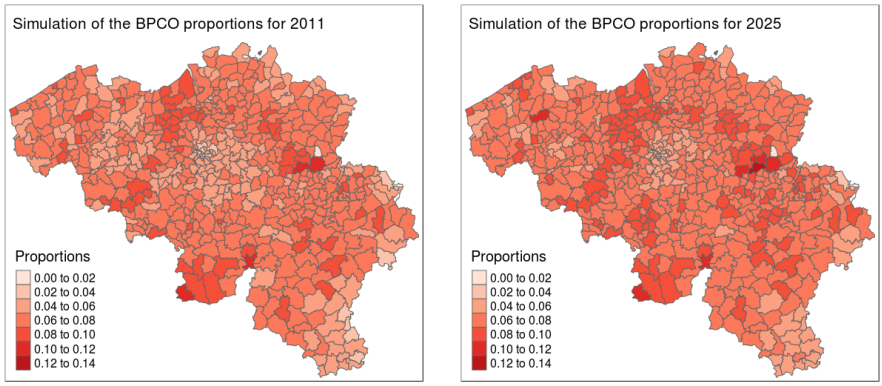


Figure 3.28 – Example of use of the platform for estimating the counts of persons with BPCO for 2011 and 2025

for the Walloon Region that will be able to use the platform for their simulations. Note that the simulated population is still ageing, thus implying a necessity to prepare the health system to an older average population. This can be verified on Figure 3.29.

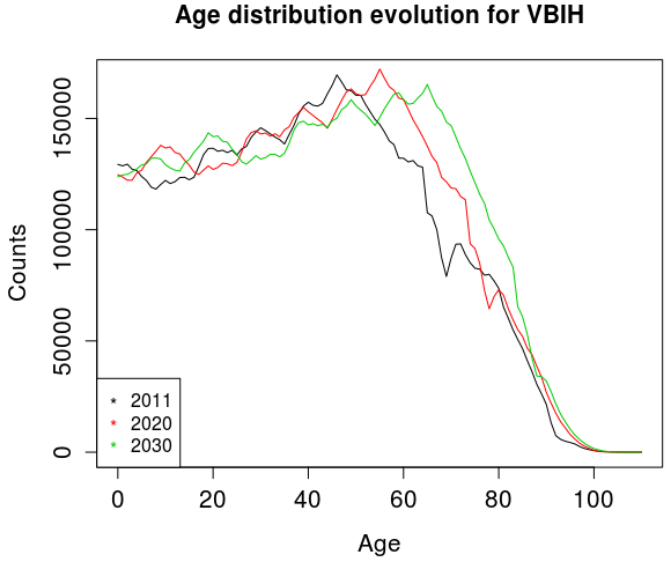


Figure 3.29 – Simulated evolution of the age distribution

## 3.6 Conclusion and discussion

In conclusion, we create a powerful framework software grounded on an agent based model including several modules for the evolution of the population, with a synthetic population as initial population. Each module is implemented independently. Some of them are obtained applying simple probabilistic methods, while divorces and marriages are more elaborated. Our algorithm seems reasonably stable and the results are not apparently too much influenced by the seed of the random number generator. However, a further stability analysis could be performed to better quantify the impact of the seed on more parameters. Furthermore, a complete sensitivity analysis could be performed, even if consequent, since there are a lot of input parameters. When the next census will be available, comparing the simulation for this year and the real census would help to quantify the efficiency of the platform.

The applied method is globally simple and each sub-part (the synthetic population generation or each module) could be improved and investigated in more details independently from the other ones. However, this simulation delivers reasonable results and the resulting platform allows the Walloon Region to be independent for their future simulations. Note that this work has been done from 2014 to 2017 and based on data of 2011.

It is fundamental to notice that these kind of simulations hides a lot of assumptions and are just estimations of what could happen if the assumptions hold, but no one could pretend to guess the future. In conclusion we hereby propose a set of projections of the Belgian population up to 2030 based on the knowledge of the initial population and on a set of assumptions.

The real advantages of the developed framework are first that we have a complete access to the whole process and thus the power to change everything we would like to test. Secondly, invoking a synthetic population simplifies the potential problems of privacy. Finally, the municipality and household considerations open the possibility to perform a wide range of future analysis with health data of different types. This flexibility is really appreciable.

This application is different from previous chapter in several aspects. VBIH contains a dynamic evolution in time and space, whereas the microsimulation step in chapter 2 considers a fixed moment (the end of the deterministic dynamic model). Moreover, to develop VBIH, census data provide us with exhaustive data at a fixed moment, whereas exhaustive data is impossible for the space debris case.

The development of this framework software raises several methodological questions that are discussed in some following chapters, such as the order of the dynamical modules or the way to model very imbalanced classification problems (such as for the divorces). VBIH acted as a seed for the different research questions which were further investigated within the core of this thesis.

# Part III

# Methods

Chapter 4

# Order of the procedures of the dynamical evolution

## 4.1 Introduction

Using agent-based model to simulate the evolution of a statistical population consists generally of two major steps, as it the case for VBIH (Chapter 3), each of them having its own set of challenges :

1. **the generation of the synthetic population:** the goal of this step is to generate a baseline population of agents which is statistically as similar as possible to the population of interest. The synthetic population generation has been extensively studied in the literature in the last two decades since the seminal work of Beckman et al. (1996). Many methods and algorithms have been designed depending on the available data for the generation process (Gargiulo et al., 2010; Barthélemy and Toint, 2013; Huynh et al., 2016; Ye et al., 2017; Dumont et al., 2017*c*). We refer the reader to (Lenormand and Deffuant, 2013; Lovelace and Dumont, 2016; Ye et al., 2017) for a review of existing approaches as well as their performances and drawbacks.

2. **the dynamic evolution of the population:** in this step, the dynamic evolution of the baseline population of agents is simulated in order to forecast the future population. This is done by defining a set of models, rules and interactions for the agents. A large number of agent-based models aiming to reproduce the evolution of a population have been developed over the years, such as ILUTE (Miller et al., 2004), MOBLOC (Cornelis et al., 2012), VirtualBelgium (Barthélemy, 2014) and its extension VirtualBelgium in Health (Chapter 3) and TransMob (Huynh et al., 2015).

The second step usually involves many different models. For instance, we can have models to simulate ageing, births and deaths in the population, the evolution of

the socio-professional status (i.e. student, retired, active, inactive) and the marital status (single, married, de-facto,etc) of the individuals, their health,etc

It is clear that the ordering in which such models are executed could have a significant impact on the final forecasted population as well as other factors such as the choice of the pseudo-random number generator, its seed and the quality of the data. Hence finding the ordering which allows to produce the most accurate results is a critical issue (Dumont et al., 2017*b*). Despite its importance, to the best of our knowledge, this problem has not yet been properly investigated in the literature. Indeed the order is arbitrarily fixed in every application, without detailing why a particular order has been retained. This gap in the literature motivated this work, aiming at providing reasons behind the selection of a particular order over others.

Chapter 3 developed a complete framework (VBIH) forecasting the population for Belgium, with each sub-module well defined and calibrated. Moreover we assume time to evolve in discrete steps of duration one year. The order in which the procedures are applied, has been intuitively chosen, but different choices could also be justified. For this reason, the analysis of the robustness of the method to changes in the order of the sub-modules appears relevant to quantify the importance and the impact of this order. Indeed, if interchanging sub-modules influences highly the results, then the modellers should be aware of this fact and clearly inform the users of the platform of the limitations and hypotheses behind the framework.

The goal of the platform "Virtual Belgium In Health" is primarily devoted to answer a precise question from the Walloon Region, for this reason it has not yet been the object of a scientific publication but only described into internal reports. However, the interest in this topic is widespread and researchers in the group "SMART" of the University of Wollongong have developed a completely similar framework (Huynh et al., 2015), called TransMob, used to simulate the dynamics of a metropolitan area in South East of Sydney, with demographic evolution. This model is totally validated and based on a synthetic population (Huynh et al., 2016) similar to the one developed for VBIH. Therefore, the analysis of the order of the appearance of the sub-modules is performed on the latter, which contains a synthetic time evolution of the population also considering ageing, death, birth, divorce and marriage, with all probabilities depending on age.

In this chapter, we will test every feasible order of the models implemented in TransMob. The resulting populations will then be compared in order to characterise the impact of the ordering of the models. In addition the sensitivity of TransMob to the seed of the random number generator used by the models will also be tested. Finally, we will propose a method to decrease the impact of the order by randomly assigning dates of births and deaths for every individuals.

The chapter contains the results that are presented in two publications. First, the importance of the order is established in (Dumont et al., 2017*b*). Then, a calendar

based approach is proposed to respond to the sensibility of the results to the order (Dumont et al., 2018). The complete citations are :

---

M. Dumont, J. Barthelemy, T. Carletti, N. Huynh (2017), Importance of the order of the modules in TransMob [Huynh et al., 2015], *Proceedings - 22nd International Congress on Modelling and Simulation*, p 811-817

M. Dumont, J. Barthelemy, N. Huynh, T. Carletti (2018) Towards the Right Ordering of the Sequence of Models for the Evolution of a Population Using Agent-Based Simulation. *Journal of Artificial Societies and Social Simulation, 21(4)*

---

## 4.2 TransMob

This Section briefly introduces TransMob, an agent-based model for simulating the dynamics of a metropolitan area in South East of Sydney, Australia. This microsimulation integrates six major modules[(1)] interacting with each other: synthetic population generation and evolution, perceived liveability, travel diary assignment, traffic micro-simulator, residential location choice and travel mode choice. The interactions between those modules are described in (Huynh et al., 2015).

Each simulated individual, or agent, is characterised by several attributes, including age, gender, household relationship, household type, identification of the synthetic household he/she belongs to, and the identification of the census collection district the synthetic household resides in. Complete details on the generation and the attributes of the synthetic population can be found in (Huynh et al., 2016).

In this work we will focus on the models responsible for the demographic evolution within the synthetic population module. TransMob evolves the synthetic population developed in (Huynh et al., 2016) with a timestep of one year for a predefined time horizon, which is set to ten years in this work. A snapshot of the synthetic population is then generated every first of January.

The approach consists of five dynamical processes executed in this specific order: ageing, dying, giving births, divorcing and marrying. It is clear that out of these five processes, only ageing is deterministic (every individual ages). On the other hand the remaining processes are stochastic, i.e. they occur randomly depending on probabilities extracted from available data. Moreover, for death, divorces and marriages, the probability of these events are conditioned by age and gender, and the probability of giving birth is conditioned by the number of previous pregnancies and the age of the female agent. The overall procedure is illustrated in Figure 4.1. Depending on the event, the structure of the household can be updated. For additional information, these

[(1)] TransMob contains different modules, each one composed of different models.

evolution algorithms are fully detailed in (Huynh et al., 2013).

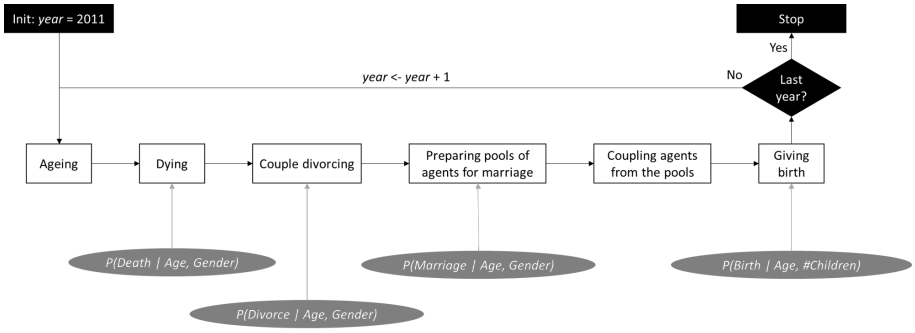


Figure 4.1 – Transmob: Flowchart of the evolutionary models.

For each simulated year, a probability for each possible event is assigned to each synthetic agent. As any other stochastic simulation, these probabilities are then used to determine which events are triggered. As these simulations are not deterministic, several runs could result in slightly different final populations. To control this, a seed can be chosen for the random number generator used by TransMob.

Note that, such as for VBIH (Chapter 3), the whole population passes through a process before moving to the next one. Focusing on individuals (or household) and running all processes for a person (or household) before considering the next one could also be possible, but it would be equivalent to what is made for the module independent of the rest of the population and, focusing on the individuals (or households) will be a problem for the modules depending on other individuals (or households) (marriages for example).

This chapter will focus on the order in which the different modules responsible for the update of the social structure, are applied in the model. The aim is to analyse the impact on the results if the order of the procedures is changed. What is the impact on the results if we decide to perform the divorces before the marriages instead of the contrary? To reach this goal, the platform has been adapted to easily handle the reordering of the modules, hereby codified with integers, "0, ..., 4", using different orders. For example, if the input is "0, 1, 2, 3, 4", the considered order is age, death, divorce, marriage and birth.

All possible combinations of orders arrangements are then associated with the permutations of the numbers from 0 to 4. Thus, 120 different orders could be analysed. However, if birth is applied before age, then in the first year, we will add the new babies to the babies already in the initial population and it will make an artificial peak of 1 year old agents the first year, 2 years old in the second year etc. For this reason, we only consider orders performing age before birth. This reduces the number of

admissible orders to 60. We will use different analyses, such as a clustering (Rokach and Maimon, 2005) and a decision tree (Breiman et al., 1984).

## 4.3 Stability

The stability of the results provided by the algorithm with respect to the randomness introduced by the stochastic processes in action was checked in Huynh et al. (2015) for the permutation chosen in this article.

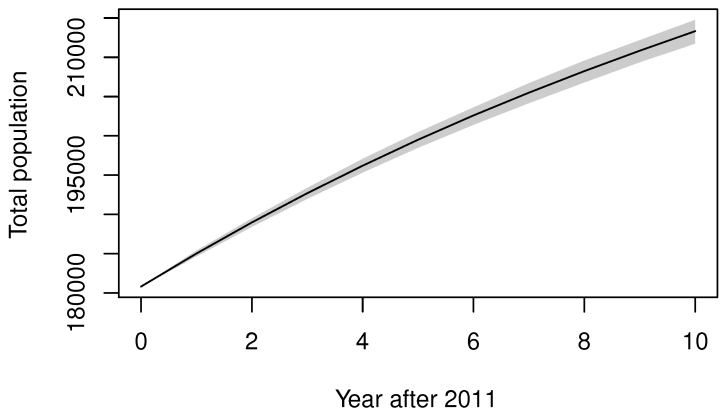


Figure 4.2 – Stability - average and ranges of the simulated population. The solid line represents the average and the shaded zone the max-min.

The first step is to confirm the stability of the algorithm also once we introduce the modules with different orders. Figure 4.2 illustrates for each chosen order in which the sub-modules are executed and over 20 seeds, the average population size as a function of time since the beginning of the simulation, expressed in years, in black and the ranges (min/max) in grey. We observe that all simulations lie very close to each other, which is a qualitative indication for the stability. The difference always increases with the number of simulated years, however let us observe that the values remains relatively small during the first ten years. The minimum, maximum and average are important characteristics, but the information of the distribution between these lines is also very useful. The number of men and women after 10 simulated years for 20 different seeds is reported on Figure 4.3. The seed does not seem to influence these two indicators.

A Bartlett test confirms the homogeneity of the variances of the results through the seeds (with a $p$-value of 0.9868 for women and 0.9065 for men). Moreover, a Shapiro test indicates that for each seed, the distribution of the number of men and women

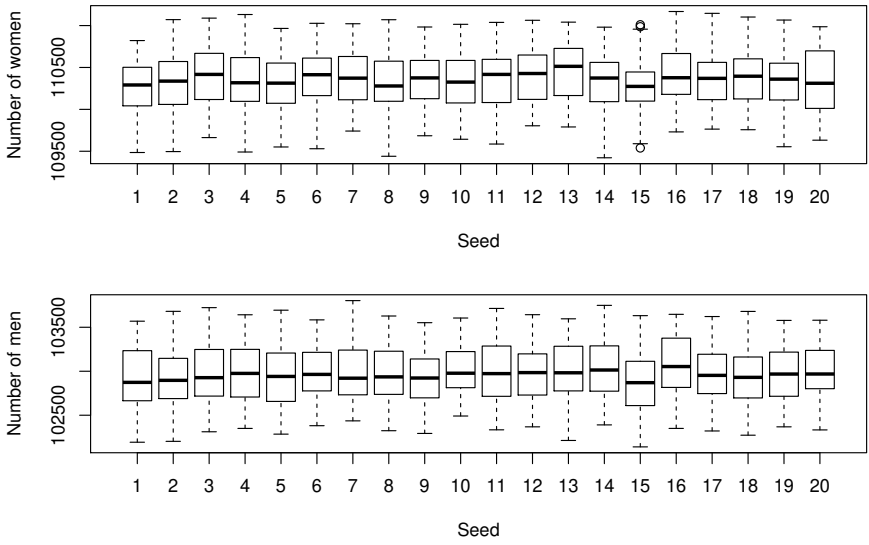


Figure 4.3 – Stability per seed for each feasible order of the processes after ten simulated years

follows a Gaussian law at level 0.01 (the smallest $p$-value, 0.04, are obtained for seeds 13 and 20, all other seed being above 0.05). A statistical test with the null hypothesis "The mean of the final population is the same for all seeds" has been executed implementing an ANalysis Of VAriance (ANOVA)(Chambers et al., 1992). The reference hypothesis is not rejected at level 0.01 ($p$-value of 0.26 for women and 0.31 for men). Thus, the seed of the random number generator does not influence the final simulated population. We therefore conclude to the stability for the generations.

We can also verify if some random seeds influence the process in a specific direction. For example, one specific seed could systematically results in an older population. Figure 4.4 contains the pair plots of the final number of women, men, less than 30 years old, between 31 and 60 years old, more than 61 years old, and the total population with a color per seed. The colors are totally mixed, indicating that no seed seems to influence these indicators.

## 4.4 Influence of the order

Does the order in which the procedures are applied influence the tendency of the results? The idea is now to determine if some orders result in a larger/younger/etc population. To analyse the influence of the order on the predictions, the output of the algorithm has been redesigned to include the order of the processes, the seed (to

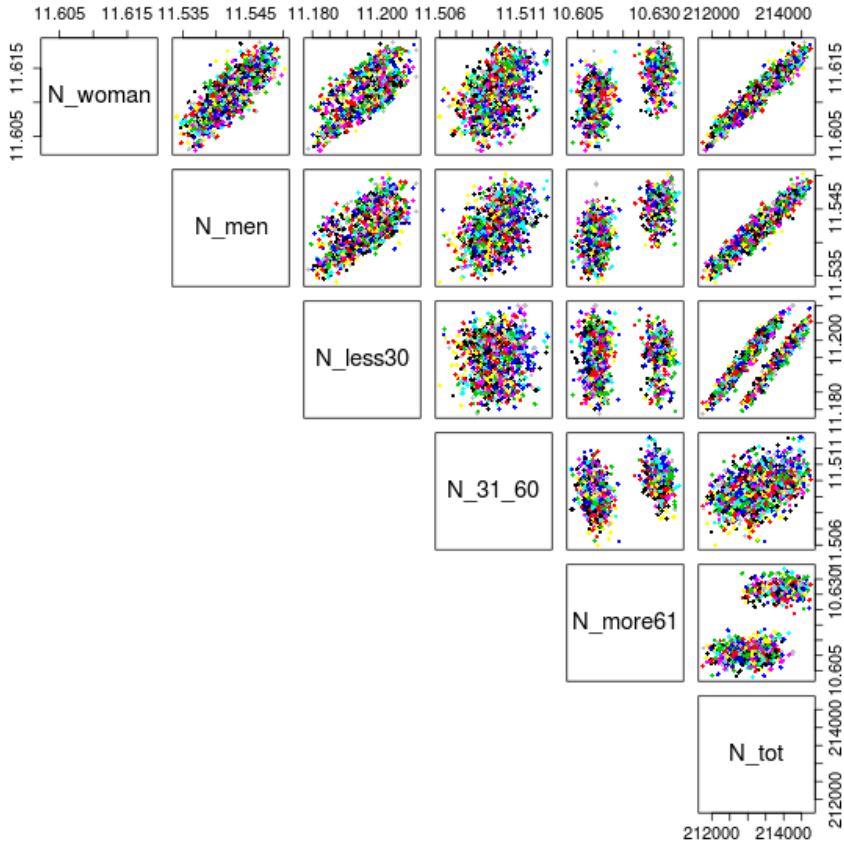


Figure 4.4 – Combinations of number of women, men, less than 30 years old, between 31 and 60 and more than 61 years old after 10 simulated years (one color = one random seed).

check that they are not determinant of the classes), and the results in 2021 (10 years of simulations).

Before analysing the influence of the order on the results, a statistical test is performed to ensure that the order significantly bias the final population. For this purpose, the homogeneity of variances of the size of the final population through each possible order is checked using a Bartlett test. The $p$-value is 0.38 and confirms this homogeneity. The normality of the final population for each order (over the 20 seeds) is validated by running a Shapiro test, resulting in 59 $p$-values above 0.05 and one of 0.038. The ANOVA being robust against the violation of this normality hypotheses and all orders accepting the normality at level 0.01, this method can be use for com-

paring the means final population per seed. The ANOVA obtains a $p$-value lower than $10^{-16}$. In conclusion, the average number of simulated individuals after 10 years is not identical depending on the chosen order of the dynamical processes.

### 4.4.1 Classification

Knowing that different orders not necessary result in the same final population, we perform a non supervised classification trying to identify trends in the results for specific orders. To identify the differences between the orders, two types of variables need to be distinguished:

1. **indicators of the final population:** the number of men, women, as well as the number of individuals in each age class (less than 30 years old, between 31 and 60, and more than 60 years old);

2. **indicators of the order:** the position of each process in the chosen order. For example, if we simulated ageing, then death, then marriage, then birth and finally divorce, the indicators of order are : position of ageing = 1; position of death = 2; etc

For the first set of indicators, the logarithm with base 10 has been taken for each variable to reduce the impact of exceptionally large populations. A clustering is performed on the indicators of the final population and the clusters are explained thanks to the order and/or the seed. The number of classes, determined thanks to the elbow method using the $k$-means clustering (Hartigan and Wong, 1979), is two (see Figure 4.5).

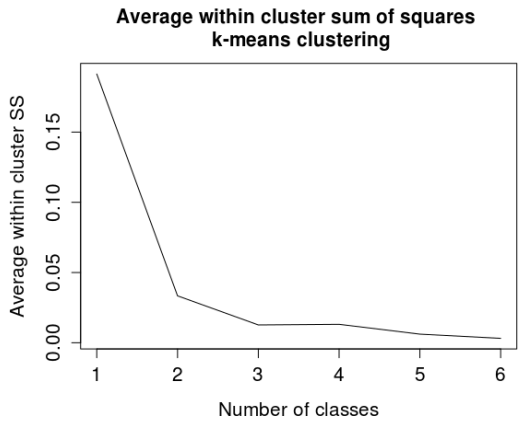


Figure 4.5 – Elbow method to determine the number of classes

The classification in two classes appear evident when performing a principal component analysis (Wold et al., 1987), or PCA for short, to visualize the population indicators. The resulting 2 and 3 first principal component illustrated on Figure 4.6, confirm the two very distinguishable set of points. The *k*-means clustering with two classes corresponds to the colors in the graph and matches with the intuitive two well separated classes. Note that the three first components computed by the PCA already explain 99,18% of the total variance. We also tried in 3, 4 and 5 classes, but two seems to be the best number, as observable on Figure 4.6.

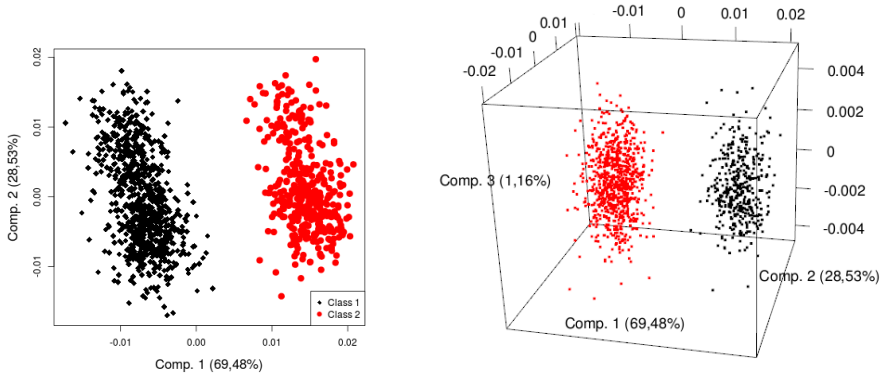


Figure 4.6 – PCA to illustrate the classification of the simulation for 20 seeds and 60 orders. Each dot represents one simulation. Two clearly separated classes can be identified.

The next step is to identify the discriminant factors for those two classes. The idea is that the classification is performed on the indicator of the final population (the number of individuals per age classes and gender). The whole set of simulations (the 60 orders over the 20 seeds) are clearly separated in two classes in terms of these indicators. Each simulation being assigned to a class, the order and the seed are now considered to identify common patterns on the simulations in the same class, in terms of seed or order of the processes. The following considers thus the problem as a supervised classification, with the indicators of the order and seed as explanation variables. We have first observed that each order over the 20 seeds always lie inside the same class. Thus, only the order categorised the simulations. Thanks to these classes, a decision tree (Breiman et al., 1984) can be constructed, with as explanatory variable the places of each process and the seed. In our case, the decision tree for two supervised classes is illustrated in Figure 4.7.

In each leaf of the tree, the number of simulations concerned by this leaf assigned to each class. For example, the first leaf above contains all simulations and we have a total of 400 and 800 simulations from first and second class respectively. The second class is then chosen by the tree for this leaf. Then, depending on the place of the

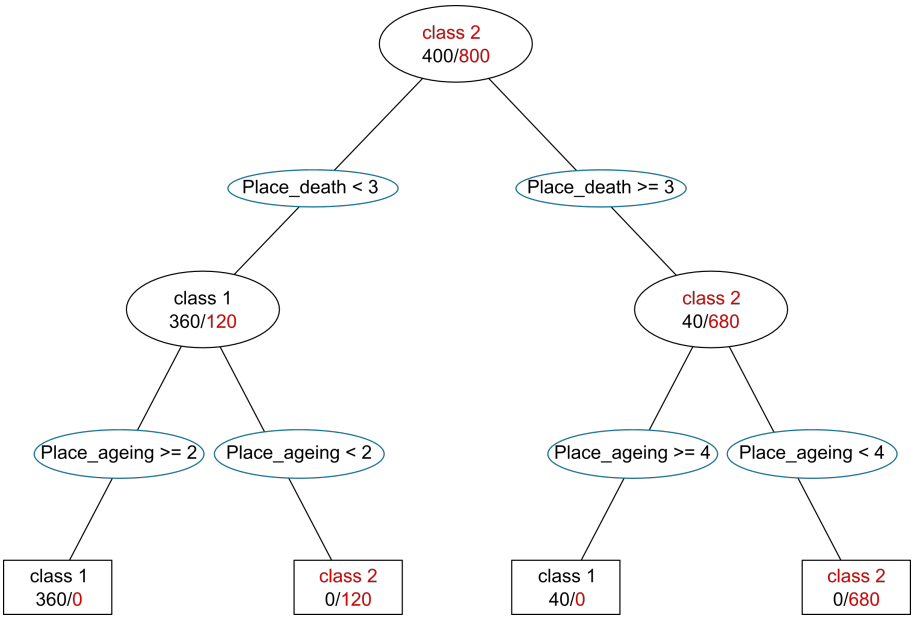


Figure 4.7 – Decision Tree with the indicators of the order of the procedures and the seed as explanatory variables

process "death", two leaf are defined. In the case of death process proceeded at the beginning of each dynamical iteration (place 1 or 2), it results in 360 simulations from class 1 and 120 from class 2, with first class thus chosen. We observe that the position of death and ageing in the process is sufficient to explain the classification, since all leaves of the last layer are pure. When we analyse more deeply each branch of this tree, we observe that the position of ageing relatively to the one of death is determinant. When ageing is before death, the simulation ranks into the second class and at the opposite, death before ageing results in the first class. Intuitively, this phenomena can be explained by the fact that the probability to die depends on the age. Indeed when ageing, the probability to die increases.

The impact of the order on the output is measured by analysing the population indicators per class. Remember, we record several indicators for the predicted population after 10 years of execution. Figure 4.8 shows that for both genders, the first class includes a larger population. Thus, when ageing is executed after death, the output of the algorithm is a larger population. This result is explained by the fact that when ageing, the probability to die becomes higher.

The results for ages are illustrated in Figure 4.9. Both classes are similar in terms of number of individuals less than 30 years old. However, the older categories are

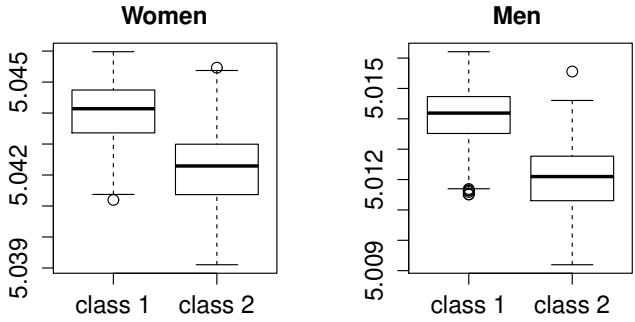


Figure 4.8 – Boxplot of gender per class (in log10-scale)

more represented in the first class. So, when ageing is executed before death, the final population is younger. Intuitively, when ageing is performed before death, the rates of death (depending from age) becomes higher and the final population contains less elderlies. It should be noted that the *k*-means algorithm begins with a random classification. For this reason, we perform it 10 times to check the validity of the results. The 10 generations gave similar results (data not shown).

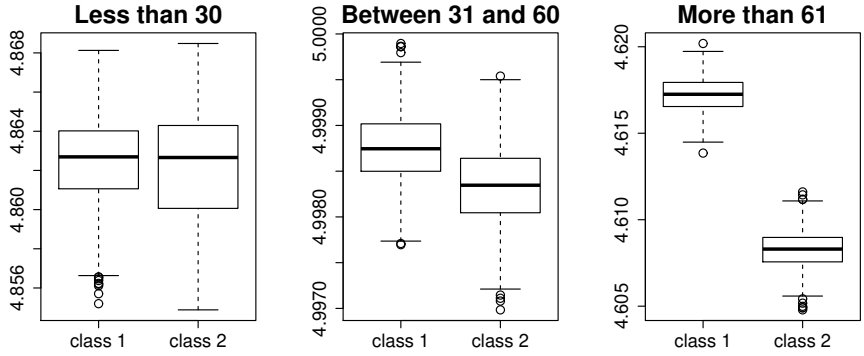


Figure 4.9 – Boxplot of ages per class (in log10-scale)

In addition, all pairs plot of the final populations per class are reported on Figure 4.10. Two well separated set of points clearly appear on this graph for each combination of indicators involving the number of individuals being more than 61 years old. On one side, the red class stands for all simulations with less elderlies. And on the other side, the black class contains populations with a larger number of elderlies. Moreover, on each graph involving the total number of individuals, we observe that the red dots tend to represent populations smaller than the black ones.

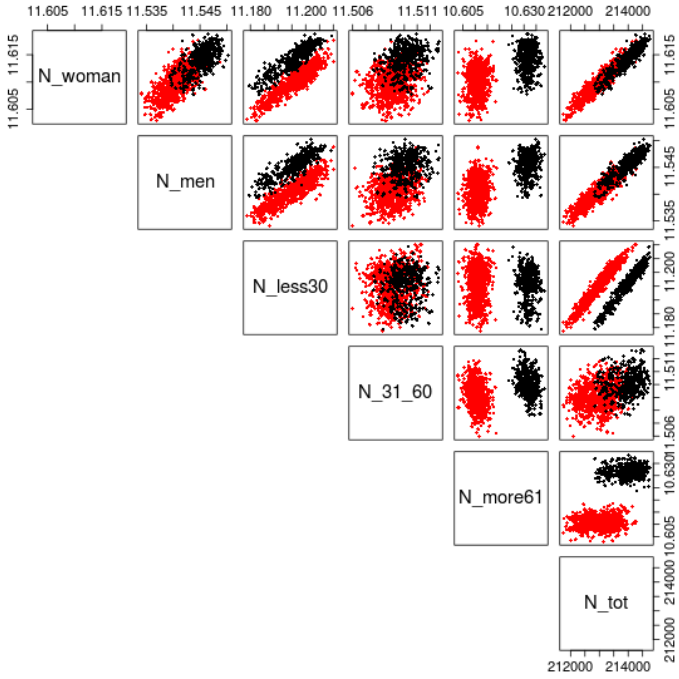


Figure 4.10 – Graph of all pairs of final population indicators per class. Each dot represents a simulation.

We can also notice two almost parallel lines for the combination of the total population and the individuals less than 30 years old, meaning that by staying on the same class, the increase of the final population implies a constant increase in the number of less than 30 years old persons. However, the two classes are well separated, indicating that for two simulations producing the same population size, populations in the red class contains more people younger than 30 years old. Similar conclusions apply when focusing on the number of individuals less than 30 years old per gender, even if this is less prominent. For individuals between 31 and 60 years old, no clear distinction can be made between the classes. In summary, the black class contains larger populations with more elderlies and less young people.

Now that the tendency of the final population is explained, we focus on the evolution of these differences through the simulated years. For this purpose, the average age of the population is calculated per gender, per year, and for each simulation (20 seeds and 60 orders). Figure 4.11 indicates, per class of the orders, the mean of these average ages and the first and third quartiles. It confirms that the first class (having ageing

before death) results in a younger population. The difference between the two classes begins early in the simulation and becomes higher from year to year. However, the curves don't really diverge and stay close to each other for these 10 simulated years. The ageing of the population is also observable on this graph, but the process tends to decrease the differences of average age between men and women in all simulations, independently of the order of the dynamical processes.

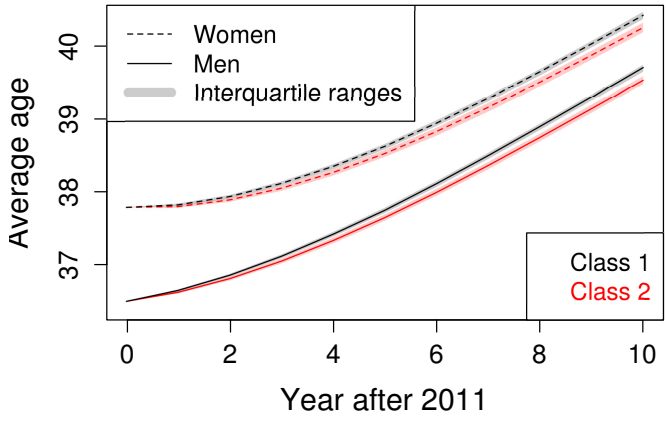


Figure 4.11 – Average age of the population per gender for 20 seeds for each possible order (classified by the decision tree)

In summary, the order significantly influences the final population. Indeed, performing ageing before *death* results in a smaller and younger population.
At this level, the position of the other processes in the dynamical evolution loop does not significantly influence the results. The following section proposes a method removing these two events from the possible orders, which enables the analysis of the impact of the order of the three remaining processes.

### 4.4.2 Correlation analysis

The *k*-means classification allows preliminary results, but is sensitive to the scale (changing some variables from number of individuals to number of millions of individuals for example could influence the results) of the data and gives only convex classes. The application of the logarithm (in base 10) on the final population indicators attempts to avoid the first potential influence, but a complementary analysis is performed, using Table 4.1, which contains the Pearson correlations coefficients between the results (after 10 simulated years) and the place of each process. To facilitate the reading of the coefficients, we coloured in red high correlations (more than 0.5 in

absolute value), in green middle high correlations (between 0.3 and 0.5) and in blue the very low correlations (less than 0.05 in absolute value). A star means that the coefficient is not significantly different from 0.

| | N_women | N_men | N_less30 | N_31_60 | N_more61 |
|---|---|---|---|---|---|
| Place_age | 0.51 | 0.55 | -0.11 | 0.31 | 0.68 |
| Place_death | -0.46 | -0.52 | 0.01* | -0.33 | -0.73 |
| Place_div | 0.16 | 0.14 | 0.23 | -0.01* | 0.01* |
| Place_marriage | -0.42 | -0.36 | -0.60 | 0.01* | 0.00* |
| Place_birth | 0.51 | 0.49 | 0.41 | 0.15 | 0.34 |

Table 4.1 – Correlations between the place of the process in the dynamical evolution and the results

The place of age and death has an important role on the final population. Indeed, when ageing is performed later, the population of more than 61 years old will be higher (correlation of 0.678). On the contrary, when death arrives later, we simulate less elderlies 10 years after. We can see that the order has a higher influence on the number of men than women. The observations on this table confirm the results obtained by the clustering, but include also additional conclusions.

Marriages seem also to have a determinant role in the results. Indeed, if marriages arrive late, we have a smaller number of young individuals. Note that when analysing the whole Java script simulating the evolution, we observe that the creators of the code have chosen that the module of birth allows only married female to have a child (note that the definition of "married" women also includes de facto relationships). For this reason, if birth is before marriages, less female can have a baby and there are less young people.

Finally, the place of birth is also important. When birth arrives later, the population becomes bigger. It is fundamental to note that we forced birth to arrive after ageing. This implies that if birth is at the beginning of the process, ageing is also at an early place. We suspect that the high coefficients for the place of birth are biased by this decision.

In conclusion, the order of the processes statistically influences the final population. The place of ageing with respect to the place of death influences the results. When ageing is before death, the final population is younger.

## 4.5 Calendar-based approach

Since the positions of death and ageing bias the simulation we decided to propose in this section an alternative method to reduce this impact (Dumont et al., 2018). By proposing another way to consider ageing and death, the possible remaining orders involve only marriage, divorce and birth, reducing the feasible orders from 60 to 6.

### 4.5.1 Method

To avoid the high influence of the position of death and ageing, our proposition is to assign a specific date for these events for each synthetic individual. For death, this means assigning a date of death and for ageing, a birthday. This technique can be easily extended to other processes as dates could be assigned to every event. This date assignment allows to use uniform or non uniform distributions if needed, since some processes are more or less frequent in a part of the year.

The proposed approach of defining dates for events to avoid problems with the possible orders is not limited to ageing and death in population evolution. It can be applied in all fields using these kind of agent-based modelling and dates can be generated for each model. We focus here on dates for ageing and death to analyse the impact on the final population since we established above that these two processes strongly influence the size and age of the final population. In our case, divorces are proposed only to couples and marriage only to single individuals. Thus these models concern only a part of the population. Even if performing one after the other can slightly modify the set of individuals going through the other model, their impact is limited.

First, the model responsible for death is executed. For each person not dying during this simulated year, the remaining models stay unchanged. However, each individual dying during this year is assigned a date of death and he/she will remain in the population, possibly performing other actions if they arise before his/her death. If an event concerning this agent is planned to happen, we check that this arises before the death date. For this, a date is randomly chosen (in a specific distribution) for this event. Moreover, the event is considered only if prior to death.

Secondly, a date of birth is also assigned to each individual. Figure 4.12 illustrates the changes induced by adding this birthday. The colors represent the probabilities of occurrence of a specific event depending on ages. We can see that the standard approach with ageing at the end (or at the beginning) considers the age at $1^{st}$ of January (or $31^{th}$ December) for the whole civil year, whereas the calendar-based approach adapts the probabilities for each individual at their birthday. This means that on one hand the standard approach changes the probabilities for everybody at the same moment. On the other hand, the calendar-based approach will change the probabilities at a different moment for each person, depending on her birthday.

It can be noted that the computational cost is totally different from adopting a timestep of a day. Indeed, a daily timestep implies considering each process for each individual for each simulated day, whereas our approach still considers each process only once a year for each individual.

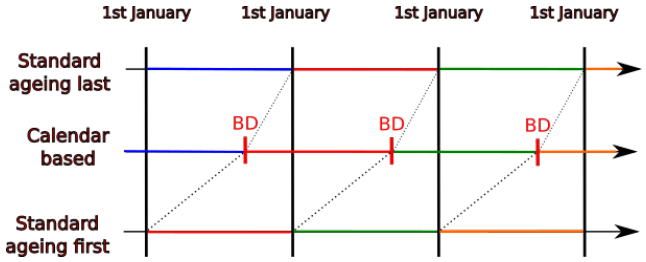


Figure 4.12 – Illustration of the addition of a birthday. Each color corresponds to an age conditioning the probabilities used by a given process.

The proposed methodology is possible only if we can establish the probability of the event occurring during the year depending on the age of the agent and its birthday.

### Naive approach

The naive approach consists in considering that the probability of an event occurring during a civil year can be refined using a convex combination of the probability of the event to happen at the present age and at the age +1. For a person of age $A$ at the beginning of the year and $BD$ days from $1^{st}$ of January to its birthday, the probability of an event $E$ occurring during the year can be calculated by

$$\begin{aligned} P(E) &= P(E \text{ only before BD} \quad \textbf{or} \quad E \text{ only after BD}) \\ &= P(E \text{ only before BD }) \quad + \quad P(E \text{ only after BD}) \end{aligned}$$

since "$E$ only before BD" and "$E$ only after BD" are disjoints. For the sake of simplicity, the naive approach is to approximate "$E$ only before/after BD" with "$E$ before/after BD" without checking that the event is not happening in the other period of the year; this assumption gets exact in the limit of events arising only once per year. Remark that this is an implicit assumption already present in the fixed order algorithm. By making the assumption that the distribution of the event occurring each day of the year is known,

$$P(E \text{ before BD }) = P(\bigcup_{i=1}^{BD} E \text{ on day } i\ ) = \sum_{i=1}^{BD} P(E \text{ on day } i\ )$$

and

$$P(E \text{ after BD }) = P(\bigcup_{i=BD+1}^{365} E \text{ on day } i) = \sum_{i=BD+1}^{365} P(E \text{ on day } i\ ).$$

In this example, we now make the simplifying but unrealistic assumption that $E$ has the same likelihood to occur any day of the year. Thus, we have[(2)]

$$P(E \text{ on day } i) = P(E|A) * \frac{1}{365}$$

with $P(E|A)$ the probability of the event for an individual during the whole year while he is of age $A$. Finally, the expression of the probability of an event during a civil year is given by:

$$P(E) = P(E|A) * \frac{BD}{365} + P(E|A+1) * \frac{365 - BD}{365}.$$

Intuitively, this splits the year into two different parts separated by the birthday, and each one having its own probability for $E$ which depends on the age. It should be noted that this makes the assumption that the probability of the event is uniformly distributed through each day of the year once we fix the age of the agent. This could be improved by approximating the probability of each day using, for example, a spline or a regression, if additional information are available. With this probability definition, ageing needs to be at the end of the process.

It can also be noted that even if we assumed a uniform distribution for the dates, we could easily use any kind of distributions for each model (e.g. an empirical distribution if the data is available). When considering a uniform distribution of the dates for an event that can arise only once per year (each day has same probability), this can be seen as a sequence of Bernoulli experiments for each day that succeeds if the event happens. The formal analytical determination of this formula gives very similar results to the naive approach developed in this section.

**Formal approach**

The aim of this section is to establish formally the probability of an event $E$ during a civil year depending on the age $A$ of the agent at the beginning of the year and its birthday happening a day $DB$.

Instead of directly computing this probability, the first step consists in considering the complementary probability of the event, i.e. the probability that the event is not happening during the year. This can be expressed as the probability that the event is not happening in any days during the year. Let us denote by $E_i$ the event $E$ occurs on day $i \in \{1, \ldots, 365\}$. If we assume the conditional independence between the $E_i$, we have:

[(2)]This also assumes P($E$ on day i and not on another day)=P($E$ on day i).

$$\begin{aligned}
P(E \mid A,BD) &= 1-P(\neg E \mid A,BD) \\
&= 1-P(\bigcap_{i=1}^{365} \neg E_i \mid A,BD) \\
&= 1-\prod_{i=1}^{BD} P(\neg E_i \mid A) \prod_{i=BD+1}^{365} P(\neg E_i \mid A+1) \\
&= 1-\prod_{i=1}^{BD} (1-P(E_i \mid A)) \prod_{i=BD+1}^{365} (1-P(E_i \mid A+1))
\end{aligned}$$

This general expression holds for any distribution of the independent events $E_i$. In our context, we make the additional assumption that the events in the set $\{E_i \mid i = 1,\ldots,BD\}$ are identically distributed, as well as the events in the set $\{E_j \mid j = BD+1,\ldots,365\}$. Thus we can write:

$$\begin{aligned}
P(E \mid A,BD) &= 1-\prod_{i=1}^{BD} (1-P(E_{BD} \mid A)) \prod_{i=BD+1}^{365} (1-P(E_{365} \mid A+1)) \\
&= 1-(1-P(E_{BD} \mid A))^{BD} (1-P(E_{365} \mid A+1))^{365-BD}
\end{aligned}$$

Using a similar reasoning, the probability $P(E_{BD}|A)$ can now be derived thanks to the probability $P(E \mid A)$ provided in the input tables and using the fact that the $E_i$ are independent and identically distributed. Indeed, we have:

$$\begin{aligned}
P(E \mid A) &= 1-P(\neg E \mid A) \\
&= 1-P(\bigcap_{i=1}^{365} \neg E_i \mid A) \\
&= 1-\prod_{i=1}^{365} P(\neg E_i \mid A) \\
&= 1-\prod_{i=1}^{365} (1-P(E_i \mid A)) \\
&= 1-\prod_{i=1}^{365} (1-P(E_i \mid A)) \\
&= 1-(1-P(E_i \mid A))^{365}
\end{aligned}$$

Isolating the probability for a specific day $P(E_i \mid A)$, we obtain :

$$\begin{aligned}
(1-P(E_i \mid A))^{365} &= 1-P(E \mid A) \\
1-P(E_i \mid A) &= \sqrt[365]{1-P(E \mid A)} \\
P(E_i \mid A) &= 1-\sqrt[365]{1-P(E \mid A)}
\end{aligned}$$

As the $P(E_j \mid A+1)$ can be obtained in a similar way, we now have all the elements to be able to generate the probabilities $P(E \mid A, BD)$ required by the model.

To illustrate the relation between the exact and naive probabilities depending on the birthday, two examples have been plotted in Figure 4.13. The differences are almost indistinguishable on the left panel representing probabilities of having a first baby at the ages of 25 and 26. On the right, a second test takes into account probabilities more affected by the change in age (difference of 20%). The difference is noticeable for the birthday on the middle of the year, but the two methods stay really close to each other. For this reason, the naive approach can be considered, since it is easier and gives similar results.

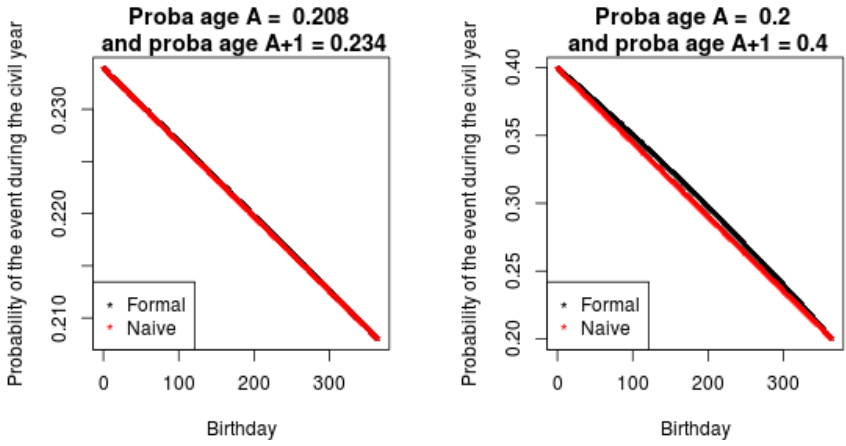


Figure 4.13 – Probabilities with the naive and formal approach when probability at first age is 0.208 and at age +1, 0.234. This corresponds to the probabilities of having a first child at age 25 and 26 respectively (left panel). And same probabilities when probability at first age is 0.2 and at age +1, 0.4 to illustrate a more remoted (right panel).

### Representation

A schematic representation of the calendar-based approach is given in Figure 4.14. The calendar based approach considers only few computational increasing since it only modify the probabilities of the other events and verify, for the individuals dying during the simulated year if the other possible events are occurring before death.

Since we change the procedures for ageing and death, the only remaining possible events are *marriage*, *birth* and *divorce*, leaving only 6 possible orders.

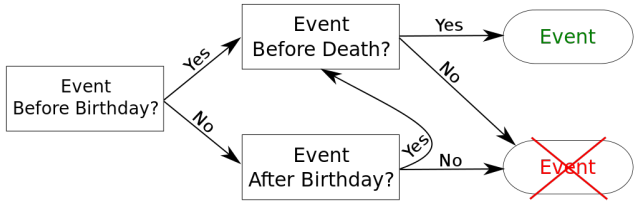


Figure 4.14 – Flowchart of the new method.

### 4.5.2 Analysis of the new orders

Similarly to the analysis presented in Section 4.4, a classification of the indicators of the final population is also performed using the new improved method. Figure 4.15 contains the elbow method to determine the number of classes showing an evident elbow at two classes.

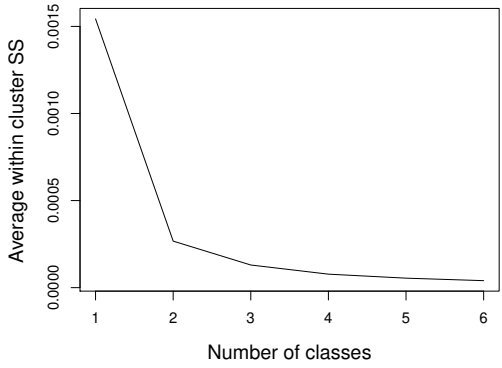


Figure 4.15 – Elbow method to determine the number of classes

These two classes are reported on the PCA in Figure 4.16. The separation of the points is less obvious than for all previous orders. Indeed, no empty space divides the two set of dots. This seems to indicate that the final populations are more homogeneous than previously. However, it is worth analysing the influence of these classes on the final population.

Figure 4.17 indicates less evident differences between the classes than for the standard method. Nevertheless, the first class (black) has a smaller population composed of less individuals under 30 years old. A slightly linear relation stands between the total population and the number of women, men and individuals less than 30 years old, meaning that the larger the total population is, the higher these indicators also are. Yet, the number of individuals older than 31 years old does not follow this linear tendency.

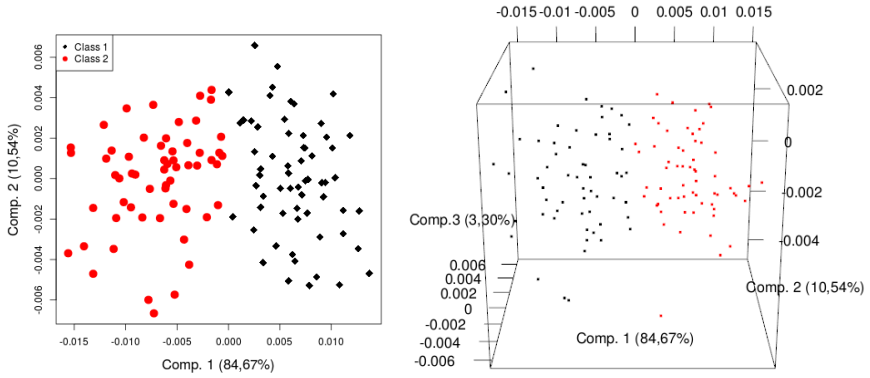


Figure 4.16 – PCA for the classification of the method with the dates. Each dot represents one simulation. The separation between the two classes is less evident using the calendar-based approach.

Identifying the patterns in the same classified orders, with the indicators of the order of procedures and the seed as explanatory variable, is the next step. A decision tree[(3)] highlighted the importance of the position of marriage regarding to the birth. Nevertheless, this pattern is less determinant than the one in the simulation without the calendar based approach. The relation between marriage and birth is very important in the model, since only married women can give birth (see (Huynh et al., 2015) for more details on models and (Huynh et al., 2016) for the definition of "married" women, which also includes de facto relationships). Having now only 6 possible orders, Table 4.2 presents the number of simulations in each class for each order in more details. One can appreciate from this Table that some orders can belong to both classes, even though there a clear tendency for one of the class. This indicates that the seed has now a larger impact than previously and supports the observation about the homogeneity of the final populations.

| First Model | Second Model | Third Model | #Simulations in Class 1 | #Simulations in Class 2 |
|---|---|---|---|---|
| Marriage | Divorce | Birth | 16 | 4 |
| Marriage | Birth | Divorce | 20 | 0 |
| Divorce | Marriage | Birth | 19 | 1 |
| Divorce | Birth | Marriage | 0 | 20 |
| Birth | Divorce | Marriage | 1 | 19 |
| Birth | Marriage | Divorce | 2 | 18 |

Table 4.2 – Classification of orders with the addition of dates

[(3)] Obtained using the package (Therneau et al., 2017).

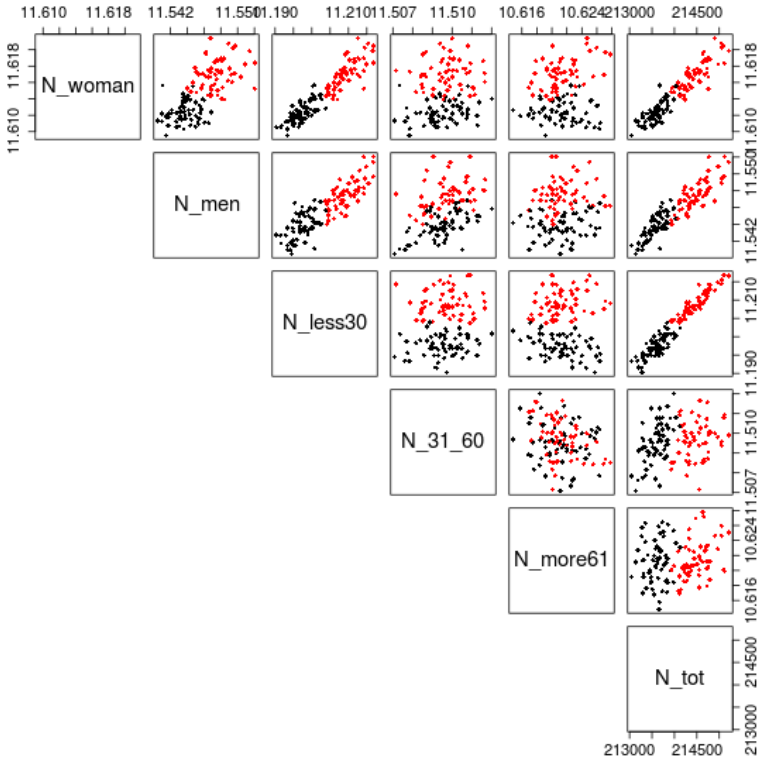


Figure 4.17 – Graph of all final population indicators per class for dates simulations. Each dot represents one simulation.

## 4.6 Comparison

In this Section we compare the performances of the calendar-based approach against the classical one. The main purpose of the new method is to reduce the variability of the final populations. For the comparison, the final population indicators after 10 years are computed for 20 random seeds and for all feasible orders with and without the introduction of dates of birth and death.

The homoscedasticity of total population indicator over the two groups *with* and *without* dates is tested. Note that the group with calendar-based contains 120 simulations (6 orders and 20 seeds), whereas the other group includes 1200 simulations (60 orders and 20 seeds). Owing to the groups being unbalanced and quite small, a careful choice of the method to test homoscedasticity needs to be done. (Parra-Frutos, 2013) analyses different statistical tests and concludes that in unbalanced and small samples, the best ways to test homogeneity of variance include the James test, the

Welch test and the Alexander and Govern test. (Dag et al., 2017) rassembles these tests in a package for the R programming language (R Core Team, 2018). The three tests allow concluding the non homogeneity of the variances with a confidence level of 0.95. The standard deviations are 645.05 for the classical simulations and 529 for the simulations using dates. This indicates that the proposed method reduces the variance between runs.

To continue, the average population and the $IQ_{95}$ interval per year and per type of simulation are depicted in Figure 4.18. The difference in variances is observable, and the difference of averages is also noticeable. The addition of dates seems to give sensibly larger populations on average, overlapping the top half of the $IQ_{95}$ of the standard simulations. The calendar-based approach tends to generate slightly larger populations with smaller ranges than the standard method.

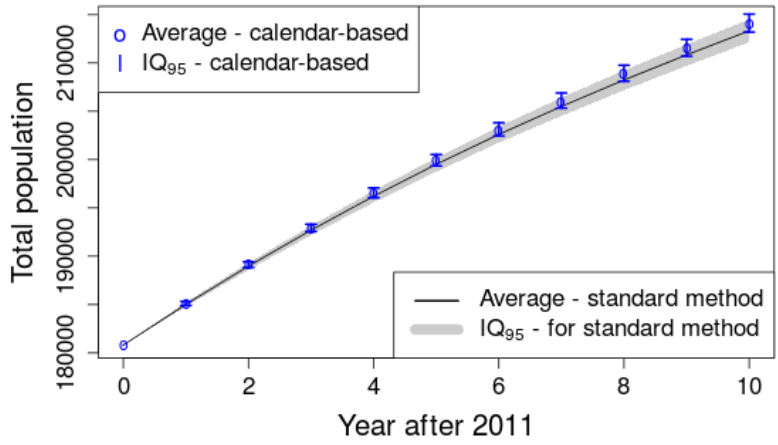


Figure 4.18 – Uncertainty analysis of standard and proposed models. Evolution of the total population for the calendar-based and classical method.

As previously stated the classification of the standard simulations divided them in two groups depending on the final population size. This observation leads us to verify if the calendar-based simulations match the class associated with the largest final populations generated by the standard approach. Figure 4.19 focuses on the 5 last simulated years and shows the $IQ_{50}$.

By analysing this graph, we can see that the use of the calendar-based approach produces final populations similar to the ones in the first class in terms of total population. Performing again the tests advised in (Parra-Frutos, 2013) to test the homogeneity of variance, we obtain for the three tests that the variances inside the first class and the simulations using dates are not significantly different.

At this point, the comparison of the distributions becomes interesting. As the

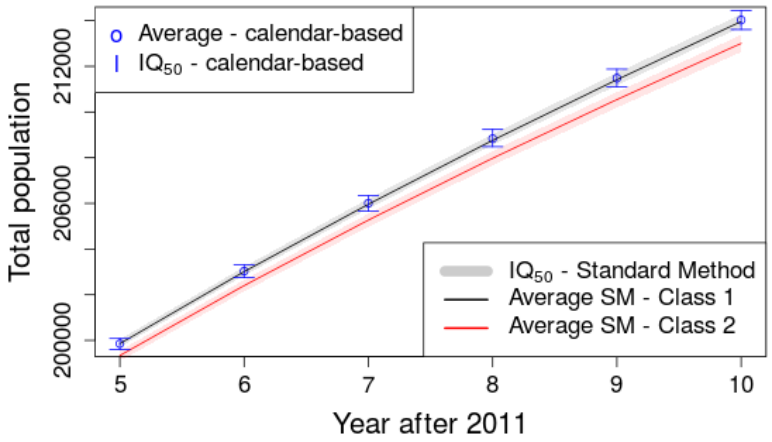


Figure 4.19 – Uncertainty analysis of classified standard and proposed models. The calendar-based approach results in populations simimar to the first class of standard simulations (ageing before death).

assumptions for the classical ANOVA test are not met, we use the non-parametric Kruskal-Wallis test. The $p$-value of 0.47 indicates that no distribution stochastically dominates the other. This is confirmed by the relative difference between the average of the two groups being only 0.03%.

Having seen that the size of the populations produced by the calendar-based approach and the first class of standard methods are statistically similar, we now look at the structure of the different populations. Indeed, the populations size could be equal while their age structure differs. This is illustrated in Figures 4.20 and 4.21 where the evolutions of deaths and births are displayed.

Contrarily to the expectations induced by the Figure 4.19, we can see in Figure 4.20 that the calendar-based approach produces a number of deaths in between the ones generated by the two classes of standard methods. This indicates that the proposed approach actually produces populations that are different from the ones belonging to the first class. The evolution of the numbers of births in Figure 4.21 tells a different story. Indeed, the calendar-based approach generates a larger number of births compared to the others two methods.

These two informations explain the fact that the calendar-based approach and the first class of standard methods generate populations of similar sizes. Consequently the proposed calendar-based methodology seems to be the most appropriate approach.

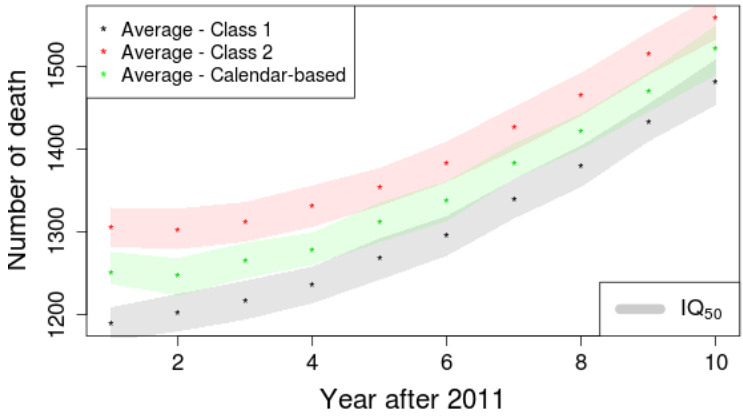


Figure 4.20 – Evolution of the number of deaths per method.

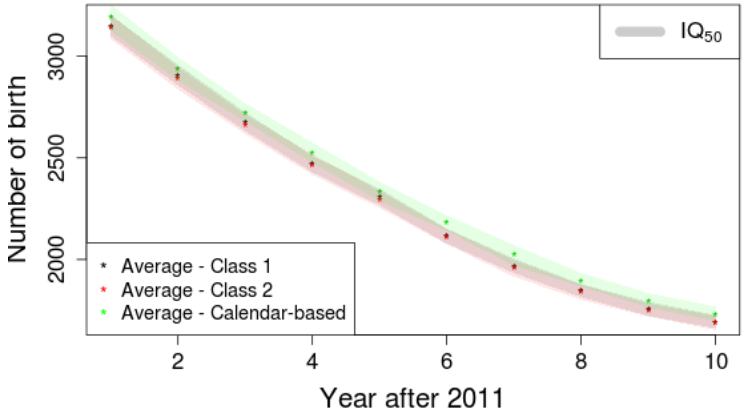


Figure 4.21 – Evolution of the number of births per method.

## 4.7 Conclusion and discussion

In conclusion, through the investigation of all the feasible ordering of models and the description of a promising calendar-based approach, this work proposes two contributions to the field of agent-based models for demographic evolutions that can be adapted for other agent-based models.

First this section showed the importance of the order of the models in agent-based modelling, after having checked the stability against random seeds. For TransMob, including five major processes: ageing, death, birth, marriage and divorce, we highlighted significant differences in the results of the simulation if death is performed after or before ageing.

Secondly, we proposed to assign dates to key events and redefine the probabilities depending on these dates. This method decreased the variability of the simulations. Furthermore it is not restricted to evolution of synthetic populations. Indeed, for each process interfering with probabilities of other model, we can assign a date for this event (either from a uniform distribution amongst the days of the year or from a defined distribution if we for example have the prevalence of birth per day in the year). Thanks to this date, probabilities of dependent events can be adapted with a weighted linear combination of the probabilities before and after the determinant event.

The proposed method allows simultaneously avoiding the bias induced by choosing an order, reducing the variability of the results and approximating a daily timestep with a reduced computational cost.

This work allows us to propose some guidelines for future agent-based models (with a discrete timestep). Indeed, for one iteration of the evolution loop, we propose the following flow:

1. Processes implying to remove agents are evaluated to identify the agents that will disappear. However, these agents are not removed directly. Instead, a moment of removal of the agent is determined.
2. Processes changing agent's characteristics that influence the probabilities of other processes are executed and timed. For individuals disappearing this iteration, we check if each event is before or after the removal moment.
3. Remaining processes are launched with updated probabilities. For individuals disappearing this iteration, we check if each event is before or after the removal moment.
4. Agents disappearing during this iteration are removed.

It should be noted that this is a general proposition limiting the influence of the order. Unfortunately, some questions could remains open. For instance, if processes are interdependent, or if we have several processes in third step, several orders are still possible.

Nonetheless there are some limitations to the proposed approach. For instance the need of additional data if one wants to draw dates from a realistic distribution through the year, the unpredictability of some events, etc. The practitioner should also be aware that all these agent-based simulations are always highly dependent to the type

and quality of input data (garbage in - garbage out process). Finally, it should also be noted that this does not necessarily allow extending the time horizon of good predictions.

These results open the question of the influence of the order chosen in Chapter 3. Indeed, the modules of VBIH are similar to the modules of TransMob and only one order has been determined. The guidelines proposed in this chapter suggest to fix birthdate and date of death for VBIH. Moreover, TransMob does not include migration modules, but the municipality often intervenes in the probabilities for the other modules. For this reason, a date of move could also be tested.

Chapter 5

# Classification with unobserved variables and unbalanced classes

This chapter is the follow up of the research associated to the modelling of the divorces in VBIH (See Chapter 3). Indeed, the problem of modelling the divorces raises a wider question, namely, how to model supervised classification processes when a class is very low represented and there are a lot of unobserved information. Note that this research has been conducted in collaboration with Dr. Johan Barthélemy from the University of Wollongong, Australia.

Classification is the process of assigning a label (or a class) to an observation given its features. It is often used in agent-based microsimulations, aiming at simulating the behaviours and states of some generic interacting entities called the agents. In such microsimulations, the core modules are designed to predict the actions conducted by the agents given their characteristics and their environment (Wooldridge, 2009). This framework has been applied in countless applications such as transportation (selecting a path to reach a destination (Barthélemy and Carletti, 2017*a*)), population dynamics (evolving a baseline synthetic population (Dumont et al., 2017*b*)), health (analysing the spread of Cholera in a population (Abdulkareem et al., 2018)), just to mention a few.

Since the set of feasible actions is often finite, each action can be associated to a label. Assigning the correct action becomes then a classification problem[(1)].

Many classification methods exist in the literature, grouped into two categories: supervised and unsupervised. Supervised algorithms require labelled training data that also provide the class of each observation. This information is then used to train (or calibrate) the parameters of the model. On the other hand unsupervised algorithms

[(1)] In this work, we will refer to label, decision, action, category and class indistinctly.

do not have access to this ground truth information and try to divide the observations into homogeneous clusters.

All supervised classification models require data containing features detailing each entity. This data allows to estimate the parameters of the model through a training process. To validate the model without using the information concerning the information of the same entities (or individuals), the data is divided into two distinct parts, one used for calibration and the other one used for validation. This allows to detect over or underfitting (Goodfellow et al., 2016).

It should be noted that in most real-life applications aiming at modelling and simulating complex agent-based systems, the decisions of the agents can not be fully explained by the observed variables or features. Indeed, there could exist many factors hidden to the modellers. The unexplained variation is then treated as a random noise which is handled differently depending on the method held by the practitioner. For instance, a linear regression assumes that the noise follows a normal distribution and explicitly incorporates it into the model formulation. On the other hand, other models, such as a deterministic neural network, do not explicitly incorporate that noise. Several models can then be applied and the selection of the best one can be a challenging question.

To find a method that would be the best for all possible applications is completely utopian, since there are plenty of methods and possible parametrisations, and testing every one in all possible configurations is simply infeasible. This work focuses on testing two simple and widely used supervised algorithms applied to predict categories or choices made by agents: *discrete choice models* and *artificial neural network*. More specifically, we will discuss the performance of the more accurate discrete choice model amongst the logit and probit versions against a shallow feedforward artificial neural network. Those methods have already been compared in previous works related to mode choice (Hensher and Ton, 2000), transport demand forecasting (De Carvalho et al., 1998), driver compliance with traffic information (Dia and Panwai, 2010), hydrogeology (Barthélemy et al., 2016) and population dynamics (Dumont et al., 2017*a*).

Our research study has been divided into two phases. First, a study directly linked to a specific application, namely to the divorces has been achieved. Then, a first attempt to generalise the results is performed, with artificially generated data. This chapter presents the two stages, after introducing both methods.

## 5.1 The methods

This section briefly introduces the two largely used methods that will be compared in this work, namely the discrete choice (probit and logit) model (DCM[(2)]) and the feedforward artificial neural network (FFNN). Let us denote by $\mathscr{O}$ and $\mathscr{C}$ the set of

[(2)] Also known as "Random Utility Model".

observations and categories, respectively. Those models can both be used to determine the probability $p_i$ that a given observation $o \in \mathscr{O}$ belongs to a category $c_i \in \mathscr{C}$ for $i = 1...m$ with $m$ being the number of categories that it is assumed to be known. More formally, this can be written as a mapping $f$:

$$o \in \mathscr{O},\ \mathscr{C} = \{c_1, .., c_m\} \rightsquigarrow p = f(o \mid \mathscr{C}, \Theta) \in \mathbb{R}^m$$

such that $\sum_i p_i = 1$, $0 \leq p_i \leq 1$ and where $\Theta$ is the set of parameters of the model that need to be estimated. The probabilities $p_i$ can be interpreted as the likelihood or confidence of a given observation belonging to each category $c_i$.

The predicted probabilities can be converted into a class value $c_k$ by selecting the class label that has the highest probability, i.e. $k = \arg\max_i p_i$. For example, let us assume that we have two classes $c_1$ and $c_2$ with predicted probabilities $p_1 = 0.62$ and $p_2 = 0.38$ for a given observation $o$. Then, $o$ will be labelled $c_1$ since $p_1 > p_2$. However, choosing a class erases the information of the certainty associated to this class. Indeed, probability of 0.51 or 0.99 for class $c_1$ gives the same result, i.e. the observation is labeled $c_1$ in both cases, even if in the first one the confidence is very close to the threshold. Another alternative is to consider the whole information about the probabilities and assign a class by performing a random draw according to the probabilities, also called roulette wheel in the framework of Genetic algorithms.

### 5.1.1 Logit and probit discrete choice models

Discrete choice models aim at explaining and predicting choices amongst a finite and exhaustive set of mutually exclusive alternatives $\mathscr{C}$ ((Ben-Akiva and Lerman, 1985) and (Train, 2009)). Discrete choice models can be considered as classifiers where an observation is an agent and the classes are the alternatives.

The key assumption of this methodology is that the agent will always opt for the alternative that maximises her utility (or benefit/cost). Let us denote by $U_{oc}$ the utility that the agent (or observation) $o$ associates with the alternative (or class) $c \in \mathscr{C}$. Then, the agent will choose the alternative[3] $c^\star$ if

$$U_{oc^\star} > U_{oc} \qquad \forall c \in \mathscr{C} \text{ and } c \neq c^\star.$$

Typically, the choice depends on many factors and some of those may remain unknown for the observer. The utility $U_{oc}$ perceived by the agent $o$ for the alternative $c$ can be divided into two parts:

$$U_{oc} = V_{oc} + \varepsilon_{oc}$$

where $V_{oc}$ and $\varepsilon_{oc}$ respectively denote the observed and hidden parts of the utility. The

[3]Note that when two alternatives have the same maximum utility, the chosen alternative is determined randomly (with same probabilities).

probability that an agent $o$ retains the alternative $c^\star$ is then given by

$$\begin{aligned} P_{oc^\star} &= P(U_{oc^\star} > U_{oc},\ \forall c \neq c^\star) \\ &= P(\varepsilon_{oc} - \varepsilon_{oc^\star} < V_{oc^\star} - V_{oc},\ \forall c \neq c^\star) \end{aligned}$$

Those equations highlight two important characteristics of discrete choice models: only the difference in utility matters, i.e. not the absolute value, and the overall scale of the utilities is irrelevant[(4)]. The former implies fixing a utility to an arbitrary value, typically 0, and the associated alternative is referred to as the base alternative.

In our context, and as it is often the case, we will assume that the observed utilities have the form of a linear model, i.e.

$$V_{oc} = \beta_c^T x_o \tag{5.1.1}$$

where $x_o$ and $\beta_c$ are two vectors, the former containing the values of the observed variables for the agent $o$ and the latter containing the model coefficients (or parameters).

Different specifications of the random components $\varepsilon_{oc}$ lead to different models (Train, 2009). For instance, we can assume that those terms are independent and follow an identical standard Gumbel distribution. This choice is widely used and allows to simplify the form of each alternative probability. Other choices could also been tested (Li, 2011), but aren't implemented in this thesis. The cumulative distribution function associated with the Gumbel law is defined by:

$$F(\varepsilon_{oc}) = e^{-e^{-\varepsilon_{oc}}}.$$

It can be shown (Train, 2009) that the probability associated with each alternative is given by (Ben-Akiva and Lerman, 1985):

$$P_{oc} = \frac{e^{V_{oc}}}{\sum_k e^{V_{ok}}},$$

which is the well-known logit model.

The maximum likelihood method can be used to determine the vectors $\hat{\beta}_c$ (one per alternative) estimating the coefficients $\beta_c$ in Equation (5.1.1):

$$\begin{aligned} \hat{\beta}_c &= \arg\max_{\beta_c} \mathscr{L}(\beta_c) \\ \hat{\beta} &= \arg\max_{(\beta_1,\ldots,\beta_m)} \sum_o \sum_c \mathbb{1}_{oc} \left( \beta_c^T x_o - \ln \sum_z e^{\beta_z^T x_o} \right). \end{aligned}$$

where $\mathbb{1}_{oc} = 1$ if agent $o$ chooses the alternative $c$ in the training data and 0 otherwise and $\hat{\beta}$ concatenates the coefficients of all alternatives.

(4) Multiplying each utility by the same factor $\alpha$ does not change the ordering of the utilities

Probit models assume that the random components $\varepsilon_{oc}$ follows a multivariate normal distribution, allowing more flexibility when modelling preferences variations within a population. In this case, the maximum simulated likelihood is also used to estimate the vector of parameters (Train, 2009).

### 5.1.2 Feedforward Artificial Neural Network

Feedforward artificial neural networks are a class of supervised machine learning algorithms inspired by the biological neural networks. The network is composed by many neurons, each one receiving a stimulus (or input) from the linked neurons, process it and forward the outcome to the following neurons if the signal is strong enough. It has been shown that neural networks can approximate any function (Cybenko, 1989; Hornik et al., 1990), hence this approach has been used in countless applications, including classification. An extensive introduction to this methodology can be found in (Kriesel, 2007), (Ripley, 2007), (Basheer and Hajmeer, 2000) or (Agatonovic-Kustrin and Beresford, 2000). (Goodfellow et al., 2016) also details why multi-layer networks are preferable to a network with only one hidden layer containing a large number of neurons.

Typically an artificial neural network is organised in interconnected layers of neurons. More specifically the feedforward architecture is characterised by one input layer, one or more hidden layer(s) and one output layer. This architecture can be described by a vector $v = [v_1, ..., v_n]$ where each component $v_i$ corresponds to the size of the layer $i$, i.e, the number of neurons in that layer, $v_1$ corresponds to the input layer and $v_n$ to the output layer. Furthermore each neuron in a layer $k$ is connected to each neuron in layer $k+1$ as illustrated in Figure 5.1, i.e. we only consider fully connected neural networks.

Denoting by $r_j^k$ the state of the $j^{th}$ neuron of layer $k$, each neuron state $r_j^{k+1}$ in the next layer $k+1$ first transforms the $r_i^k$ values received from the previous layer with a weighted sum resulting in a temporary value:

$$t_j^{k+1} = \sum_{i=1}^{v_k} w_{i,j}^k \times r_i^k + \theta_j^{k+1}$$

where $w_{i,j}^k$ are the weights of the connections of the neurons $i$ (in layer $k$) with the $j$ one (in layer $k+1$) and the term $\theta_j^{k+1}$ is the bias added to the neuron state $r_j^{k+1}$. This summation is then followed by a non-linear activation function $g$ to obtain the final value at $r_j^{k+1}$. In our context, $g$ is the *ReLU* function, which is the default recommendation when designing modern artificial feedforward neural network (Goodfellow et al., 2016). It follows that $r_j^{k+1}$ is given by:

$$r_j^{k+1} = g(t_j^{k+1}) = \max\{0, t_j^{k+1}\}.$$

In order to compute the probability that a given input belongs to the category $i$, the

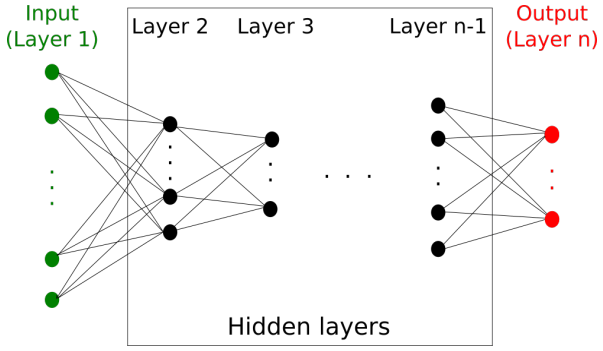


Figure 5.1 – Illustration of the architecture of a neural network composed by $n$ fully interconnected layers. The neurons in the first layer (in green) receive the inputs and are connected with every neuron of the second layer. The information progress from each layer $k$ to the next one $k+1$. The last layer (in red) contains the outputs of the neural network. Between the input and output layers, there are $n-2$ hidden layers.

softmax function is applied to the neurons in output layer, i.e.:

$$p_i = \frac{e^{r_i^n}}{\sum_k e^{r_k^n}}.$$

The values of the weights and the bias are calibrated by minimising a loss function. The specification of the function depends on the task to be performed by the artificial neural network. In the case of classification, the loss function is given by the cross-entropy defined by:

$$L(\hat{p}, c) = -c \ln \hat{p} - (1-c)\ln(1-\hat{p})$$

where $\hat{p}$ is the vector of predicted probabilities and $c$ is an indicator vector where $c_i = 1$ if the observation belongs to the category $i$ and 0 otherwise.

It can be noted that compared to the discrete choices models, this method belongs to the framework of deterministic classifiers, because the noise is not explicitly taken into account. Moreover, with the chosen specifications of both models, the form of the probability function is completely similar between the DCM using the observed utility and the FFNN the state of the neurons in the output layer.

## 5.2 Modelling the divorces

The application from which the idea of this chapter emerged is the case of the modelling of the divorces for the framework VBIH (see Chapter 3). This is equivalent to a supervised binary classification since every year each couple can decide to stay together or divorce, meaning that the couples are spread within two sub-groups. This

section aims at highlighting, for a specific application (simulating the divorces for the married couples in Belgium), the importance of the unobserved variables on the results of two types of simple yet widely used models: feedforward neural networks (FFNN) and logit discrete choice models (LDCM). The rest of this section is a slightly updated version of a part of the peer-reviewed proceeding:

---

M. Dumont, J. Barthelemy, T. Carletti (2017), Robustness of artificial neural network and discrete choice modelling in presence of unobserved variables, *Proceedings - 22nd International Congress on Modelling and Simulation*, p 480-486

---

### 5.2.1 Data

Since 1983, each citizen of Belgium is registered in the municipality she is living in, with several information about herself: gender, date of birth, address, people she is living with, marital status, etc. All these informations are gathered within a federal official database which is called the National Register. When a change occurs in a variable describing the citizen status, the latter needs to notify the municipality. There is a longitudinal recording of the data, that allows moving back in time and analysing different years or different specific path (for example, time between weddings and divorces, or the first moves after a wedding, etc.).

To model divorces we record the choice made in 2002 (either divorce or stay together) by each couple still married in 2001. In addition, each couple is characterised by : the year of the wedding, the size of household in 2001, the age class of both the husband and the spouse, the diploma level of both partners, the subjective health level of both partners and the province they are living in in 2001. This results in informations about approximately one million couples with 8000 divorcing in 2002 (corresponding thus to 0.8% of the couples divorced).

To test the methods, we randomly separate the database in two parts : one part is the calibration set, that will be used to calibrate both methods (FFNN and LDCM); the other part is the validation set. Both calibrated models will be tested on these data to check the prediction quality of the simulations. We split the data in 50% for calibration and 50% for validation using a simple random sampling scheme. Let us observe that usually scholars divide the database into 80-20 or 70-30 for the training-validation phases, however we decided to keep a consequent part in the validation set, because we have very few divorces in the whole data. Indeed, this means only approximately 4000 divorcing couples in the validation to have the possibility to check the attributes of the simulated divorcing couples.

### 5.2.2 Modelling divorces using logit discrete choice modelling

A logit discrete choice modelling is developed to simulate the divorces in Belgium. The used data are discussed in Section 5.2.1. Note that this setting results into a binary logit since there are only two possibilities (exhaustive and mutually exclusive): divorce or stay together. Since only the difference of utility matters, we decide to fix the utility of staying together to zero.

The used model being linear on the variables, several operations on the input variables have been tried. First, we recategorised each ordinate variable into integers. For instance age's intervals has been numbered from 1, for 15-19, up to 17, for 95-100 and more. We also normalised the input data (dividing each variable by its maximum). In a second phase, each variable has been transposed into boolean variables (1 if the attribute holds true, 0 on the contrary). This specific operation allows capturing non linear relations. This preliminary process shows that the model is improved when considering age of the woman in a boolean transposition, whereas it removes the significance of all categories of the age of the man. Therefore, the age is considered as boolean for the wife, but not for the husband. To calibrate and validate discrete choice models, we use BIOGEME (Bierlaire (2003)).

The selected model has an adjusted $R^2$ of 0.94, which means that the model fits well the calibration data. Note that a couple is characterised by several objects that occurs in the model: a woman ($W$), a man ($M$), an household ($HH$) and a marriage ($Mar$). The concerned characteristic is written as an index. For example $M_{age}$ is the man's age. Remember that the age of the woman is considered as boolean, implying that $W_3$ stands for a woman of age between 25 and 29 (class 3). After first inserting all possible variables and removing the non significant ones, the final model is (remember $U_{stay} = 0$):

$$\begin{aligned} U_{div} &= 2.04W_3 + 1.78W_4 + 1.72W_5 + 1.65W_6 + 1.27W_7 + 0.64W_8 \\ &\quad -0.56W_{10} - 1.01W_{11} - 1.67W_{12} - 1.83W_{13} - 0.41W_{14} \\ &\quad -0.12M_{age} - 0.13M_{dipl} - 0.08W_{subj-health} - 0.14HH_{size} - 0.02Mar_{length} \end{aligned}$$

To facilitate the interpretation of this utility, all positively influencing terms are written at the beginning of the expression. We observe that the model found a linear correlation between the age category of the man in the couple and the proposition of divorce with a negative coefficient and a non linear relation for the age of the woman. Indeed, the coefficient changes from one age category to another. We can also note that the health of only the wife and the diploma of only the husband enter into account. Moreover, with the years of marriage and the number of children, the utility of divorce decreases. This model is used to simulate the choices of the couples in the validation data. The probability to divorce is calculated for each couple. Then, 500 simulations are generated and their outcome are compared to the real choice of this couple.

Figure 5.2 shows the distribution of the number of divorce resulting from the 500 simulations. We can see that each run contains at least 3600 divorces and no more

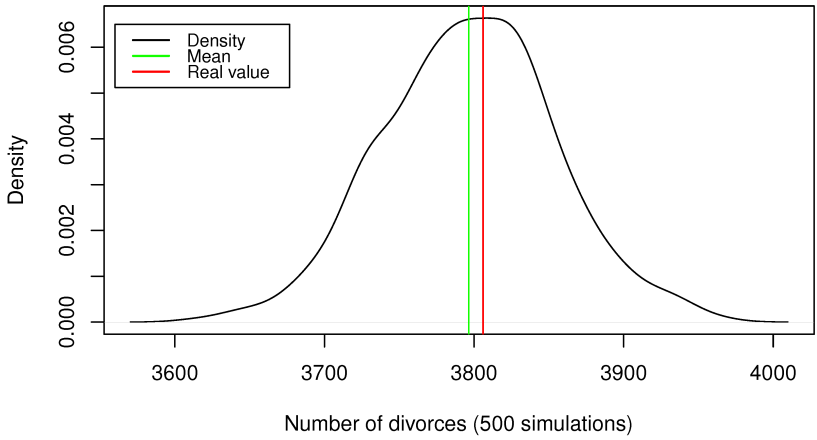


Figure 5.2 – Density of the number of simulated divorces using the discrete choice modelling (500 simulations)

than 4000, with a mean of 3797, which is very close to the real number of divorces in the validation data set. The Z-test checking if the proportions of divorces are similar in the average simulation and in the reality has a $p$-value of 0.9222 meaning that the proportion of couples choosing to divorce is well defined in the model.

Figure 5.3 presents how the number of divorces is distributed among the age classes of wife and husband (class 3 is 25-29 years old, etc), where the actual number of cases are represented in black and the model results are shown in grey (the average number amongst the 500 generations). The results shown in Figure 5.3 confirm that the actual pattern is well conserved in the model results as both black and grey bars are approximately of the same height. A t-test with as null hypothesis "The differences between the real and the average number has a mean of 0" confirms this intuition ($p$-value of 0.91 for women and 0.87 for men).

### 5.2.3 Modelling divorces using feedforward artificial neural network

To generate the results, we use the R-package *neuralnet* (Fritsch et al., 2010). Figure 5.4 illustrates the resulting network for the divorces.

The inputs are the normalised attributes of the couples and the choice is a boolean output deciding if this couple divorces or not. One hidden layer of 2 nodes is included and the threshold for the activation function is shown in blue. When using this network to forecast the divorces for the validation data, the correct choice is made for 99,2% of the couples.

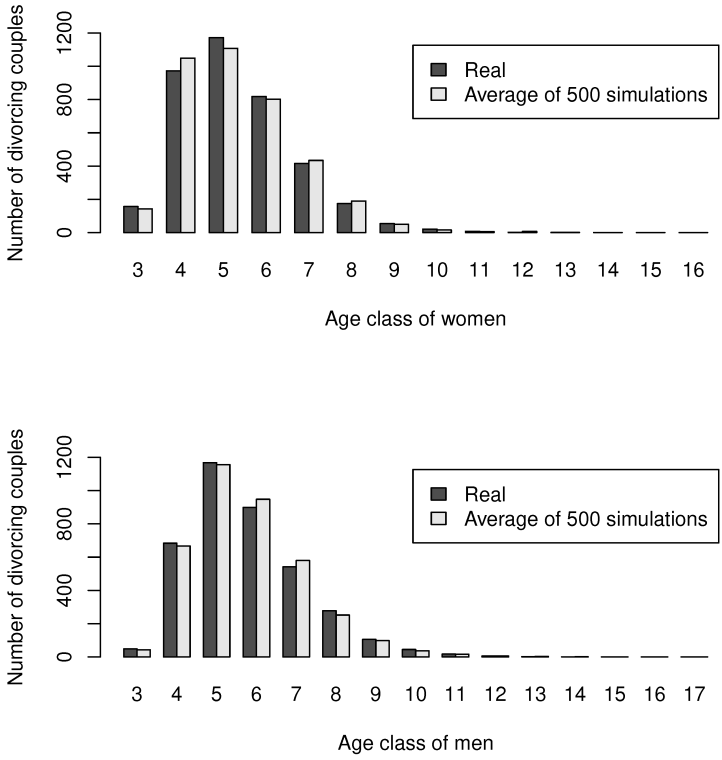


Figure 5.3 – Bar chart of the number of divorces per age class and per gender (real vs simulated)

This seems to be a very good prediction rate, but we need to go further to see if the divorcing couples have the right characteristics. Trying to analyse this, we figure out that no couples will divorce with this model. Indeed, it predicts that all couples will stay together. Since only a very small number of the couples actually divorce (0.8%), this makes a good prediction rate.

Many different configurations in terms of number of layers and/or nodes of the feedforward neural network have been tried to find a better artificial neural network, but none of them improved the results. Zur et al. (2009) shows that noise injection can reduce overfitting. So, random noise has been added to the input and to the output in the calibration step. We also tried to add a random input neuron. Moreover, to calibrate the mean and standard deviation of these Gaussian noises, a genetic algorithm was realised. However, no divorcing couples appeared.

We figure out that in the data, many couples have exactly the same attributes, but don't make the same choice. In each possible type of couple, the majority stay together

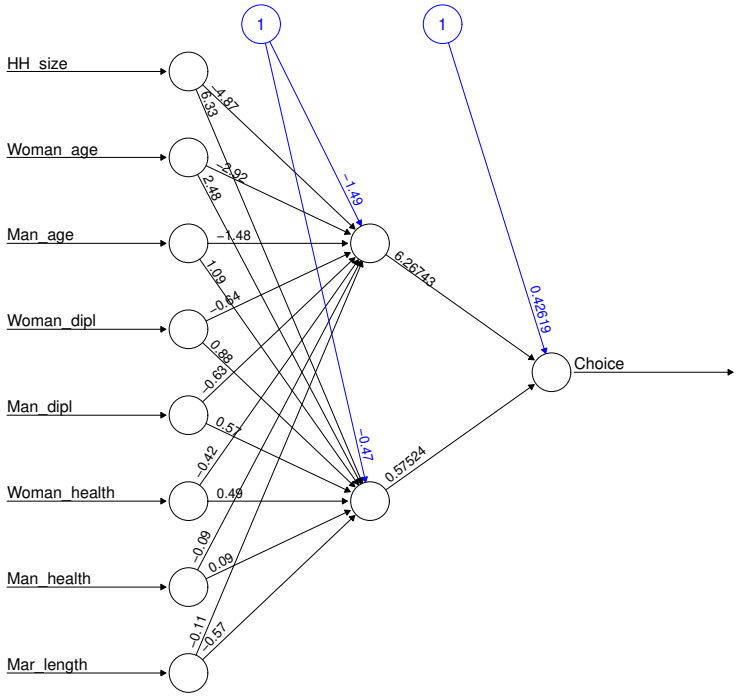


Figure 5.4 – Neural network for divorces

and a few decide to divorce. Simulations with a calibrated neural network being deterministic, it returns the same output for each similar couple. We could consider the results as a probability, because it corresponds to the proportion of divorcing couples in the category. However, a neural network is not necessary to consider proportions as probabilities.

### 5.2.4 Conclusions

In conclusion, discrete choice modelling enable us to forecast the divorces, with quite satisfactory results. Indeed, the simulated divorcing couples characteristics fit to the theoretical ones. It is a stochastic method, meaning that simulating twice doesn't necessary give the exact same results. Feedforward artificial neural networks doesn't return good results. This is caused by a presence of unobserved variables and by the fact that very few couples divorce. This method has a prediction rate of 0.99 even if it generates no divorces. Artificial neural networks give very good results in several fields, but we need to be careful once using them in presence of unobserved variables.

Forecasting the divorces is a complex task involving a wide range of factors. Some of them can be quantified (such as the number of children), but others are totally qualitative. Moreover, each individual is different and in similar contexts, two persons will not necessary make the same choice. For this reason, no completely deterministic model would well forecast this process.

A first stage to generalise this study is discussed in next sections, testing both methods on artificial data with specific characteristics, such as the presence of unbalanced categories and the presence of unobserved information.

## 5.3 Artificially generated data

To the best of our knowledge, a comparison of the performance of both considered methods (DCM and FFNN) in presence of missing variables has not been done yet, thus motivating this study. In addition, in many relevant applications, the features are unevenly shared among the classes. We are thus also interested in studying the impact of the distribution of the various categories in the data as this can have a significant impact on the training of the algorithms and the resulting outcomes. This part of the thesis aims at testing the algorithms using synthetic generated data sets, with specific characteristics (e.g. balanced or unbalanced classes) and thus deals with a completely controlled environment. And for a sake of clarity, we focus on binary classification.

Feedforward neural networks trained using back-propagation and stochastic gradient descent are known to have issues in case of unbalanced classes (Murphey et al., 2004). It is thus interesting to compare their performances against discrete choice models in those conditions, trying to identify the threshold from which the classes are too imbalanced for the FFNN.

### 5.3.1 The generated data

To investigate the impact of inputs with different characteristics, the data sets used in this work are artificially generated thanks to the function `make_classification`[(5)] of the `scikit-learn` module for the Python 3 programming language (Pedregosa et al., 2011). This method adapts the algorithm that was originally designed to generate the MADELON dataset (Guyon, 2003).

Using this tool, the generated explanatory variables will be normally distributed around the vertices of a random polytope. This method considers non independent variables[(6)]. Three different types of explanatory variables are possible: informative,

[(5)] Example of data generated by this module are available at `https://scikit-learn.org/stable/auto\_examples/datasets/plot\_random\_dataset.html`

[(6)] Note that this is different from generating each variable separately following a one dimensional normal distribution.

redundant and repeated. In this work, only informative features are created, since we aim at analysing the influence of unobserved variables, which will be simulated by removing informative variables from the training data set.

Each observation is assigned to a class. Several parameters allow us to design classes with specific characteristics. Indeed, the data can be designed with well separated or overlapping classes. Moreover, by adapting the proportion of observations in each class, unbalanced data can be generated. In this study, binary classes (True or False) are generated.

The module `make_classification` generates very different types of classes. For the sake of reproducibility, the seed of the random generator is fixed.

Let us now describe how we aim to test the influence of missing variables. Assuming that if the whole set of features influencing the decision is available, we could perfectly predict it, we generate a complete data set with only informative variables and no observation in the wrong class. All variables are informative meaning that if some variables are not used in a model, they play the role of missing variables. The data is generated around a random polytope.This artificial data generation enables the possibility to analyse the performance of both DCM and FFNN methods in different configurations:

- different proportions of unbalanced classes, defined by $p_T$, the proportion of observation in the class "True",
- different proportions of missing informative features, denoted by $p_V$,

Note that 100 explanatory informative variables are generated, all having the same range. For this study, 1000 observations are generated.

Finally, the observations are randomly split into a training and a validation data sets: 75% of the observations will be used for the training and the remaining 25% will be used for the validation.

### 5.3.2 Performance indicators

This section introduces the performances indicators used to compare the FFNN and DCM methods when the proportion of used features evolves from 0.01 to 1.00.

In order to assess the quality of the models' predictions, we need to use some metrics. We have first some indicators specific to the method. To quantify the quality of calibrated neural networks, the final loss value is considered. Indeed, the aim of the training is minimising this loss. For calibrated discrete choice models, the final log-likelihood is analysed. Indeed, training a discrete choice model consists in maximising the likelihood, equivalent to maximising the log-likelihood.

In addition, the probability of the actual class (in the training data) is also taken into account. For example, let us consider, in the validation data, an observation $x$ belonging to the class *True*. According to the classifier, the probability of $x$ being *True* is 0.46 while the predicted label is *False* with probability 0.54. This indicator is calculated for each observation and each simulation. We can then register the mean, median, standard deviation (and other statistics) of these probabilities.

### 5.3.3 Numerical experiments

Our performance quantifiers being defined, we can turn to present the numerical experiments, divided into three subsections. Both methods requiring calibration and several hyper-parameters, two first subsections present the calibration of both algorithms. Then, in the third subsection, the best found artificial feedforward backpropagated neural network is compared to the best found discrete choice model.

#### 5.3.3.1 Calibrating the neural network

In the experiments illustrated in the next Section, we will test simple feedforward architectures with one and two hidden layers. The Python 3 module `scikit-learn` has been used to optimise the parameters of the neural network.

For the artificial neural network, the structure and the training algorithm have to be chosen. Indeed, the number of layers and associated number of neurons needs to be previously determined. Then, the calibration is performed by minimising the loss function. Three different optimisation algorithms are tested : the stochastic gradient descent (SGD), another gradient-based optimization with adaptive moment estimation (Kingma and Ba, 2014) (ADAM) and quasi-Newton methods that approximates the Broyden-Fletcher-Goldfarb-Shanno algorithm (LBFGS).

We first test one unique hidden layer with a number of nodes ranging in the interval 1 to 20. These configurations are tested with 1 to 100 input variables inserted and for proportion of observations in first class ranging from 0.5 to 0.95 with steps of 0.05. A set of 1000 individuals are generated with same characteristics 3 times and the choice of included variables (from 1 to 100) is performed 5 times, randomly and without replacement. For this algorithm aiming at minimising the loss, the mean loss is considered over the different simulations. Results are similar independently of the proportion of observations in each class and of the number of hidden variables. For this reason, Figure 5.5 presents the global means (in semi-logarithmic scale). We can see that the optimizer LBFGS outperforms its concurrent for all number of nodes. Moreover, the larger the number of nodes in the hidden layer the better is the obtained classification. However, the curve is convexly decreasing meaning that the addition of a node becomes less and less interesting.

We focus now on two hidden layers. Note that LBFGS still completely outperforms the two other optimisers. This is explained by the fact that LBGS does not

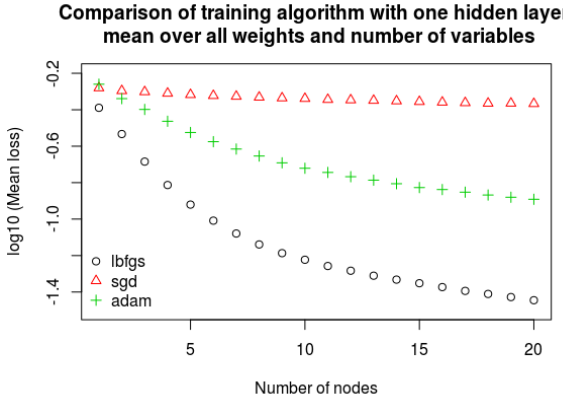


Figure 5.5 – Comparing the training algorithm with one hidden layer

belong to the stochastic gradient descent family of optimizer, thus it does not only consider a batch of the observations during the backpropagation, but rather the whole training data set and is thus well suited for small problems. For this reason, LBFGS is considered in the following. The final loss (in semi-logarithmic scale) per number of nodes in each layer is presented in Figure 5.6.

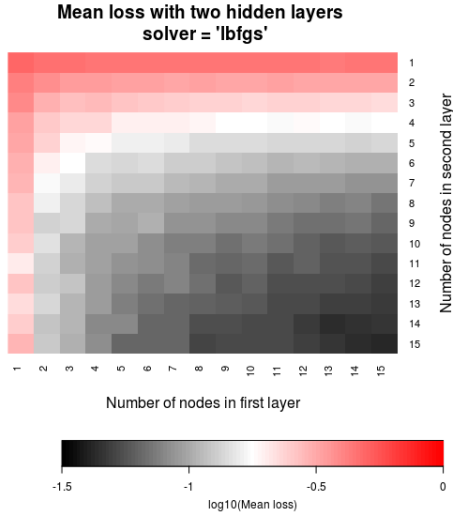


Figure 5.6 – Comparing the loss with two hidden layer, depending on the width of each layer

The more nodes included in the two layers the better the neural network. However, the mean final loss with one hidden layer containing 20 nodes is 0.036 whilst the one for two hidden layers containing both 15 nodes is 0.042. Option of one hidden layer seems thus better. Note that we tested also all possible structures with 15 to 20 nodes on each two layers and the best mean loss was 0.032 for 20 nodes on each layer. However, this option (two hidden layers of 20 nodes) doesn't give results better enough to justify the complication of the model. Note that for the same level of mean loss, a link appears between the number of nodes in each two first layers. This relation, at first glance, seems quadratic. This could be investigated more deeply in future works.

Here, the shallow feedforward backpropagated neural network with one hidden layer including 20 nodes is thus chosen for the following (optimized with LBFGS).

#### 5.3.3.2 Calibrating the discrete choice model

The function `Logit` of the Python 3 module `statsmodel` (Seabold and Perktold, 2010) has been used to perform the estimation.

Focusing on discrete choice modelling with linear utilities with respect to the parameters, the choice between optimizer and type of model (Probit or Logit) needs to be done. The same data as for calibrating the neural network has been used and tested over the two types of models for seven different training algorithms[(7)] : Newton, BFGS, CG, Newton-CG (Nocedal and Wright, 2006), LBFGS (Byrd et al., 1995), Powell (Powell, 1964) and basinhopping (Olson et al., 2012). Discrete choice calibration aims at maximising the log-likelihood function. For this reason, the quality considered here is the final log-likelihood. Moreover, the differences in the likelihoods being small, the mean probability of the real class is also taken into account.

Results are really similar in almost all cases. Figure 5.7 contains the two quality indicators per type of model and optimiser in the most differentiated case (few variables and high imbalanced classes). Logit seems better for all training algorithms for both indicators. Moreover, there is no significant differences between the optimizers.

Therefore, the chosen discrete choice model considers the logit option optimised with the training algorithm LBFGS. Indeed, logit performs better and all optimiser being equally efficient, the same one as for the FFNN is selected.

#### 5.3.3.3 Comparison of performances

Neural network and a discrete choice model being configured, performances between the two methods are compared. Both models run on exactly the same data. For this analysis, we continue with 1000 observations and 100 explanatory variables. For balance proportion $p_T$ from 0.5 to 0.975 with step 0.025, five different data sets are gen-

[(7)] All detailed in `https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html`

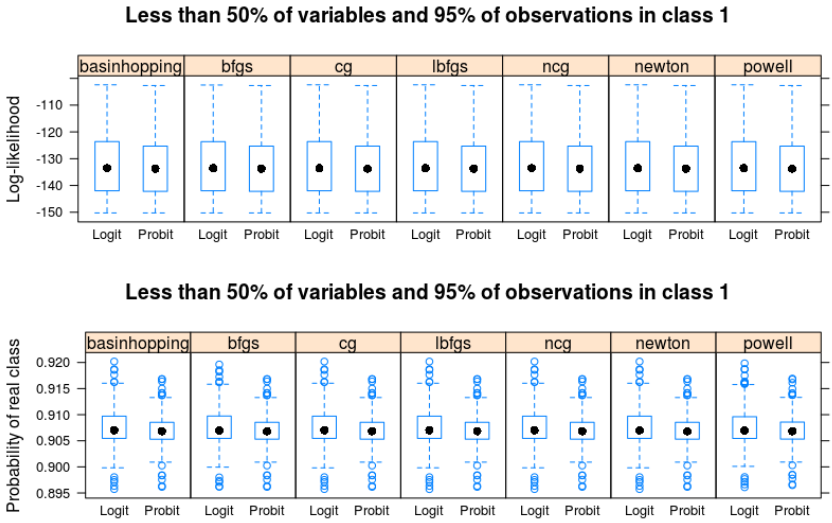


Figure 5.7 – Comparing the log-likelihood and the predicted probability of the real class depending on type of model and optimizing algorithm

erated. Then, for each possible proportion of inserted informative variables $p_V$ (from 0.01 to 1 with step 0.01), 10 different sub-data (of $p_V \times 100$ variables) are randomly chosen without replacement. Both methods are then performed for each determine sub-data.

Figure 5.8 illustrates for each simulation (corresponding to a dot) the mean probability of the real class (in validation data) for FFNN and DC. Note that if both methods give similar results, the trend should be around the identity line (added in black on the graph). FFNN completely outperforms DCM for the vast majority of the simulations. For balanced classes (proportion of 0.5), discrete choice model gives quite bad results with probabilities around 0.5 (similar to a random guess). The graph has a surprising shape with approximately horizontal lines (one per balance proportion). For few simulations, discrete choice seems slightly better, but the balance proportion doesn't explain in which conditions this happens.

Figure 5.9 shows the same results, but coloured depending on the number of inserted variables in the simulation. With all variables, FFNN has a high tendency to be efficient whereas DCM is better with only few informative variables for calibration (meaning many hidden variables). Interpreting Figures 5.8 and 5.9 simultaneously, one can see that the proportions explains the differences in the quality of the DCM while the number of variables influences more the quality of the FFNN.

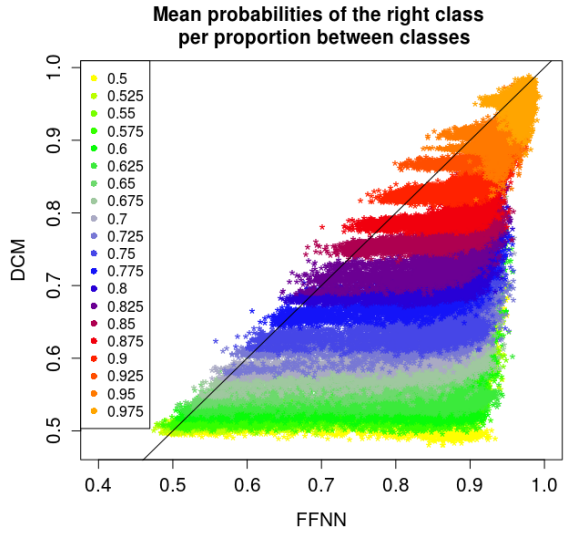


Figure 5.8 – Comparing probabilities of the real class for FFNN and DCM per balance proportion between classes.

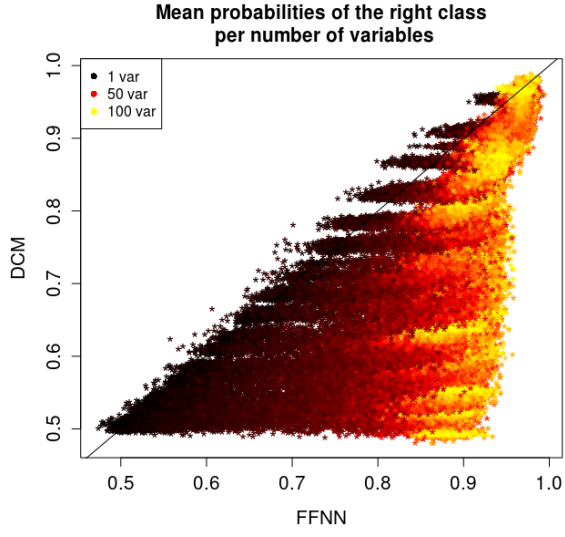


Figure 5.9 – Comparing probabilities of the real class for FFNN and DCM per number of inserted variables.

Considering for each method both information at the same time (the proportion and the number of variables), we obtain Figures 5.10 and 5.11. On one hand, DCM seems to be very sensitive to the balance of the classes, being more efficient for high unbalanced classes. On the other hand, FFNN seems to be really influenced by the presence of hidden variables. With too many missing variables, and reasonably balanced classes, it doesn't perform better than an arbitrary guess of probability 0.5. The scale is adapted to be the same on the two heatmaps allowing comparisons. It is obvious that DCM is less efficient in the vast majority of cases.

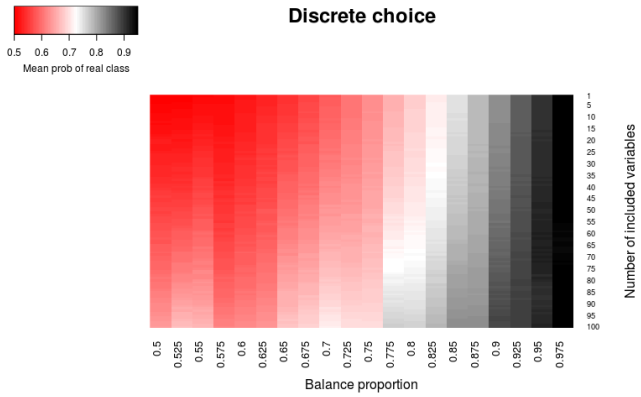


Figure 5.10 – Mean probabilities of real class depending on the number of inserted variables and the balance proportion between classes for discrete choice model.

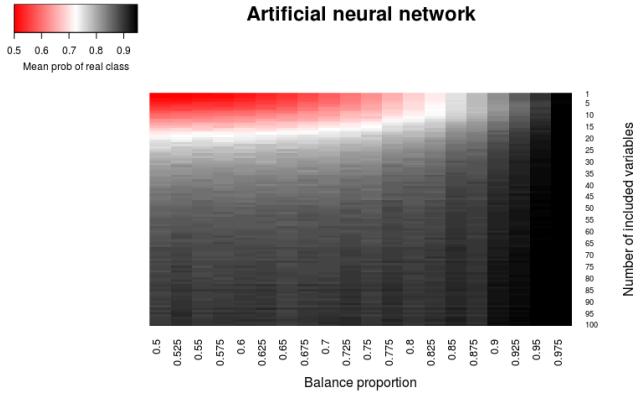


Figure 5.11 – Mean probabilities of real class depending on the number of inserted variables and the balance proportion between classes for artificial neural network.

To choose the best method (between the two tested in this chapter) in every specific case, the difference between the probabilities is performed on Figure 5.12. The white zone implies similar results for both methods. FFNN is the best in the black zone and DCM in the red zone. One can clearly see that except when having many hidden variables and a high unbalanced problem, the shallow feedforward backpropagated neural network gives more satisfactory results than the logit discrete choice model. Even if FFNN is known to be affected by unbalanced classes, without any missing variables, FFNN stays better or similar to DCM in terms of mean probabilities of the best class.

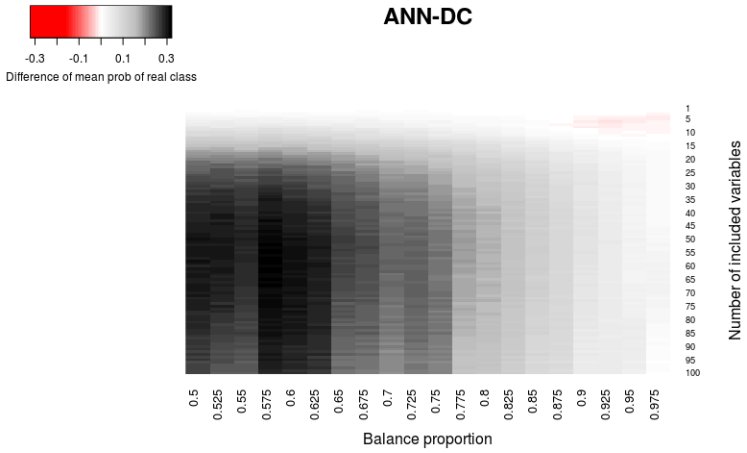


Figure 5.12 – Difference between FFNN and DCM in terms of the mean probabilities of real class depending on the number of inserted variables and the balance proportion between classes.

The standard deviation of these probabilities of the right class is also recorded for each simulation. Figure 5.13 illustrates the average of these standard deviations depending on one side of the proportion of observations in class 1 and on the other side on the number of inserted variables. Very unbalanced classes (over 0.95) signify similar standard deviations. However, before this, the probabilities of FFNN vary more than the ones of DCM. For the influence of the number of considered variables, DCM has a monotone curve, always increasing, meaning that adding variables induces a higher variability of the probabilities. Note that this is explained by the fact that for few variables, DCM is similar to a random choice, so probability close to 0.5 for every observations. FFNN behaves totally differently. Indeed, from 0 to 15 variables, the addition of an informative column results in a higher variability of the predicted probabilities for the real class. However, beyond this point, the standard deviation, so the variability is decreasing.

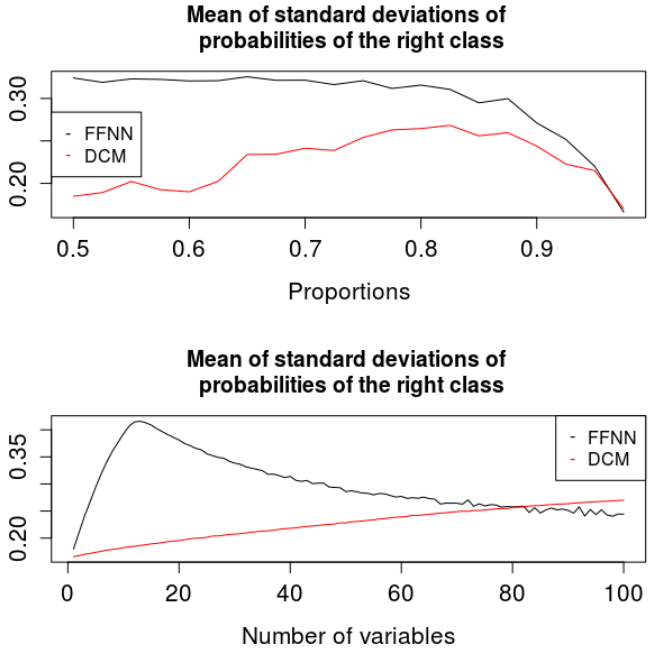


Figure 5.13 – Mean over the simulations of the standard deviation (among all observations) of the probabilities of right class depending on the balance proportion between classes and the number of inserted variables.

### 5.3.4 Conclusion and discussion

This work has investigated the impact of two factors on the prediction performance of the discrete choice model (Logit and Probit) and the feedforward backpropagated artificial neural network for a binary classification task: the number of missing variables simulating imperfect information and the distribution of the individuals amongst the two classes.

The numerical experiments support the claim that the neural network should be favoured most of the time, except when the classes are highly unbalanced and the missing information is considerable. The results are consistent with the previous section and with (Dumont et al., 2017*a*) where a discrete choice model was retained for simulating divorces within a population. Indeed, evaluating for each married couple and each year the decision to stay married or to divorce implies to train a model using an highly unbalanced data set. Moreover, it is obvious that the available variables are not sufficient to simulate this decision.

As this kind of situation arises quite often, this work can be of interests for researchers wondering which of these methods to chose from. Indeed, by checking how unbalanced the data is, the practitioner can then select the right approach.

Important conclusion is that even if FFNN is known to be affected by unbalanced classes, without to many missing variables, FFNN stays better or close to DCM.

This work could be extended by considering more classes, additional supervised classification methods (Logistic regression, Support Vector Machine, decision tree etc), testing other structures of the FFNN, different size of populations and performing additional test using other synthetic or real data sets.

The differences between the discrete choice model and the neural network have been already analysed in different scenarios in the literature, but the added value here is to consider proportion of missing variables and very unbalanced classes. These unbalanced classes induce very rare event difficult to predict, since their impact are small when measuring the quality of the result. For this reason, adapting the algorithm to add weights to each observation could be more investigated by adding costs to misclassification.

This study is helpful for dynamic microsimulations, since a lot of possible modelled events could be of rare occurrences, as for example the divorces simulated in Chapter 3.

# Part IV

# Discussion

Chapter 6

# Conclusion and perspectives

## Conclusion

In conclusion, this thesis explored the microsimulation cosmos from different perspectives. First, the power of the framework and the possibility of various applications have been proven. Indeed, we created a synthetic population of space debris, allowing to tackle the impact the latter can have on functional satellites. The use of microsimulation allowed to improve and encompass the weaknesses of the well established deterministic model. This interdisciplinary collaboration gave satisfactory results and open a wide range of possible future works. The microsimulation is performed after the deterministic simulations to adapt the results to the observed data. The space debris microsimulation is thus fixed in time and space. This application is particular, because no exhaustive data exist and therefore the simulation of small space debris is a really challenging task. On the other hand, the tool Virtual Belgium In Health has been developed. It aimed at providing the Walloon Region with a tool forecasting the health needs of elderlies, grouped into households and localised by their municipality. Contrarily to the first application, census provides exhaustive data at a point of time and the microsimulation is dynamic, evolving simultaneously on time and space (with the migrations). The coherence between the output of our microsimulation and the macrosimulation of the official office "Bureau Fédéral du Plan" has been highlighted. The tool delivered to the Walloon Region allows to add new diseases for example (available for example per gender, age and municipality) and to create maps of approximated evolution from 2011 to 2030, relying on the structure of the simulated population. The tool delivered to the Walloon Region should be simple to be used and its capabilities somehow restricted, however the potential of the dynamic synthetic population built for this project is huge. Indeed, adding new characteristics and insight about novel dynamical processes is very easy. Furthermore, the final synthetic population enables innovative possible analysis, such as for instance, estimating the proportion of households with, at least, one member affected by dia-

betes.

The second part of the thesis investigated some methodological concerns directly induced from the developed applications. On the one hand, while determining the order in which applying the different sub-models for the dynamical evolution of VBIH, the impact of the chosen order was unknown. For this reason, the design of the global organisation of the different sub-models used within the framework of a discrete time microsimulation for the temporal evolution of a synthetic population is discussed. We quantified the impact of each specific order in which these sub-models are applied in the framework TransMob, globally similar to VBIH, but without migrations. We identified two different types of outcomes according to the used orders: larger and older simulated populations are obtained when the death process is performed before the ageing and smaller and younger simulated populations result otherwise. To bypass this problem, we propose to fix dates within the sub-models of deaths and ageing and call this method the "calendar based approach". And the improvements achieved using this new method are shown.

On the other hand, modelling each possible event for the microsimulation implies to adapt the method to the considered event. For example, the divorces process, modelled, in VBIH, thanks to a feedforward backpropagated neural network or a discrete choice model, raised questions about the binary classification with unbalanced classes and/or many unobserved variables. We compared these two methods on artificially generated data with different characteristics. The result is that, for the different generated datasets, the feedforward neural network performs better than the logit discrete choice model most of the time, expect when the classes are highly unbalanced and the missing information is considerable.

# Perspectives

This thesis generates a lot of different possible perspectives. First of all the handling of the two applications could be improved and continued. Indeed, for example, the space debris microsimulation could be performed using heuristics algorithms or other methods. The technique of dynamic microsimulation to replace the deterministic simulation could also be tested. The proposed calendar based approach could be implemented for the dynamical evolution of Virtual Belgium In Health. Furthermore, it would be really interesting to analyse the reasonable forecasting time horizon and the evolution of the variation amongst the different runs through time. The important output it could raise is to determine if the variability continues growing and at which pace or if it levels off.

Furthermore, alternatively to the calendar based approach, we could test to apply the evolution processes in a random order changed every year, or to apply all processes to a household before moving to the next one (this approach is called "case-based" in-

stead of "time-based"). This technique would be easy for the processes involving only one household independently from other households, but for the marriages, this could be less evident. We could then imagine, for the marriages creation, to create a pool of "candidates to get married this year" and to create only at the end of the simulated year the couples. Note that intuitively, reducing the timestep (of one year for VBIH) also reduces the impact of the order. Testing a dynamic microsimulation with a timestep of a month for example and comparing the results to the calendar based approach could also help to move forward to less variability within the outputs.

Moreover, the most fruitful perspective, planned as one of our future research objective, consists in removing the problem of the time scale and simultaneously the influence of the ordering, by constructing an algorithm implementing the time evolution in a continuous way. These kinds of continuous time simulations are already developed to stochastically simulate coupled chemical reactions, using the Gillespie algorithm (Gillespie, 1977). The idea is to iterate from an initial population according to two major steps: determining the time before the next event and then choosing amongst the possible events the one concerned. This approach allows the use in the models of non-constant probabilities over time taking so into account seasonality effects. However, it requires to adapt the whole method used for the temporal evolution of our synthetic population of individuals, grouped into households. The first results already obtained with this "continuous" method only applied to the sub-models implementing birth, death and ageing on a population of 15.000 agents are promising, even if we are aware that large populations with frequent events, the inter time among two consecutive events scale as the inverse of the population size, could be an issue in the method when the time until next event approaches the machine precision. Note that continuous time microsimulation has already been developed with the same global idea and implemented in an available specific programming language, called MODGEN [(1)], which could be investigated also.

Finally, the feedforward backpropagated neural network for binary classification could also be investigated in more details, by for example trying other neural structures and comparing results depending on different aspects such as the complexity of the associated network or the total number of nodes or branches. We could also consider logistic regression theory and various penalties for the optimisation of the calibration.

[(1)]This has been developed by Statistics Canada and is available at `https://www.statcan.gc.ca/eng/microsimulation/modgen/modgen`

# Bibliography


S. A. Abdulkareem, E.-W. Augustijn, Y. T. Mustafa, and T. Filatova. Intelligent judgements over health risks in a spatial agent-based model. *International Journal of Health Geographics*, **17**(1), 8, 2018.

S. Agatonovic-Kustrin and R. Beresford. Basic concepts of artificial neural network modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, **22**(5), 717–727, 2000.

J. W. Bae, E. Paik, K. Kim, K. Singh, and M. Sajjad. Combining microsimulation and agent-based model for micro-level population dynamics. *Procedia Computer Science*, **80**, 507–517, 2016. International Conference on Computational Science 2016, ICCS 2016, 6-8 June 2016, San Diego, California, USA.

D. Ballas and G. P. Clarke. Modelling the local impacts of national social policies: A spatial microsimulation approach. *Environment and Planning C: Government and Policy*, **19**(4), 587–606, 2001.

A. Banos, N. Corson, B. Gaudou, V. Laperrière, and S. Rey Coyrehourcq. Coupling micro and macro dynamics models on networks: Application to disease spread. *Gaudou B., Sichman J. (eds) Multi-Agent Based Simulation XVI. MABS 2015. Lecture Notes in Computer Science*, **9568**, 19–33, 2015.

J. Barthélemy. *A parallelized micro-simulation platform for population and mobility behaviour-Application to Belgium*. PhD thesis, University de Namur, 2014.

J. Barthélemy and T. Carletti. An adaptive agent-based approach to traffic simulation. *Transportation research procedia*, **25**, 1238–1248, 2017*a*.

J. Barthélemy and T. Carletti. A dynamic behavioural traffic assignment model with strategic agents. *Transportation Research Part C: Emerging Technologies*, **85**, 23–46, 2017*b*.

J. Barthélemy and T. Suesse. mipfp: Multidimensional Iterative Proportional Fitting and Alternative Models (http://cran.r-project.org/package=mipfp). 2015.

J. Barthélemy and P. Toint. Synthetic Population Generation Without a Sample. *Transportation Science*, **47**(2), 266–279, 2013.

J. Barthélemy, T. Carletti, L. Collier, V. Hallet, M. Moriamé, and A. Sartenaer. Interaction prediction between groundwater and quarry extension using discrete choice models and artificial neural networks. *Environmental Earth Sciences*, **75**(23), 1–14, 2016.

I.A. Basheer and M. Hajmeer. Artificial neural networks: fundamentals, computing, design, and application. *Journal of microbiological methods*, **43**(1), 3–31, 2000.

R.J. Beckman, K.A. Baggerly, and M.D. McKay. Creating synthetic baseline populations. *Transportation Research Part A: Policy and Practice*, **30**(6), 415–429, 1996.

M. E. Ben-Akiva and S. R Lerman. *Discrete choice analysis: theory and application to travel demand*, Vol. 9. MIT press, Cambridge, USA, 1985.

M. Bierlaire. Biogeme: a free package for the estimation of discrete choice models. *in* 'Swiss Transport Research Conference', number TRANSP-OR-CONF-2006-048, 2003.

F. Bourguignon and A. Spadaro. Microsimulation as a tool for evaluating redistribution policies. *The Journal of Economic Inequality*, **4**, 77–106, 2006.

L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and regression trees*. CRC press, 1984.

L. Brown and A. Harding. Social modelling and public policy: Application of microsimulation modelling in Australia. *Journal of Artificial Societies and Social Simulation*, **5**(4), 2002.

R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on scientific computing*, **16**(5), 1190–1208, 1995.

D. Casanova, A. Petit, and A. Lemaître. Long-term evolution of space debris under the J2 effect, the solar radiation pressure and the solar and lunar perturbations. *Celestial Mechanics and Dynamical Astronomy*, **123**(2), 223–238, 2015.

D. Casanova, C. Tardioli, and A. Lemaître. Space debris collision avoidance using a three-filter sequence. *Monthly Notices of the Royal Astronomical Society*, **442**(4), 3235–3242, 2014.

A. Celletti, C. Efthymiopoulos, F. Gachet, C. Galeş, and G. Pucacco. Dynamical models and the onset of chaos in space debris. *International Journal of Non-Linear Mechanics*, **90**, 147–163, 2017.

J. M. Chambers, A Freeny, and R. M. Heiberger. *Statistical Models in S*, chapter Analysis of variance; designed experiments. Wadsworth & Brooks/Cole, 1992.

P.-A. Chiappori. The theory and empirics of the marriage market. *Annual Review of Economics*, **12**(1), 547–578, 2020.

S. Cho, T. Bellemans, L. Creemers, L. Knapen, D. Janssens, and G. Wets. Synthetic population techniques in activity-based research.

H. Cohn and M. Fielding. Simulated annealing: Searching for an optimal temperature schedule. *SIAM Journal on Optimization*, **9**(3), 779–802, 1999.

E. Cornelis, J. Barthélemy, X. Pauly, and F. Walle. Modélisation de la mobilité résidentielle en vue d'une micro-simulation des évolutions de population. *Les cahiers scientifiques du transport*, **62**, 65–84, 2012.

R. Costa, T. Eggerickx, and J.P. Sanderson. Les territoires de la fécondité en Belgique au 20ème siècle. *Espace populations sociétés [En ligne]*, **2011**(2), 353–375, 2011.

H. Cowardin, P. Anz-Meador, and J. A. Reyes. Characterizing GEO Titan IIIC Transtage Fragmentations Using Ground-based and Telescopic Measurements. *in* S. Ryan, ed., 'Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference', p. 36, 2017.

G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, **2**(4), 303–314, 1989.

O. Dag, A. Dolgun, and N. Konar. *onewaytests: One-Way Tests in Independent Groups Designs (https://CRAN.R-project.org/package=onewaytests)*, 2017. R package version 1.5.

M.C.M. De Carvalho, M.S. Dougherty, A.S. Fowkes, and M.R. Wardman. Forecasting travel demand: a comparison of logit and artificial neural network methods. *Journal of the Operational Research Society*, **49**(7), 717–722, 1998.

G. de Menten, G. Dekkers, G. Bryon, Ph. Liégeois, and C. O'Donoghue. Liam2: a new open source development tool for discrete-time dynamic microsimulation models. *Journal of Artificial Societies and Social Simulation*, **17**(3), 9, 2014.

G. Dekkers, H. Buslei, M. Cozzolino, R. Desmet, J. Geyer, D. Hofmann, M. Raitano, V. Steiner, P. Tanda, S. Tedeschi, and F. Verschueren. What are the consequences of the awg-projections for the adequacy of social security pensions? *SSRN Electronic Journal*, 2009.

N. Delsate and A. Compère. Nimastep: a software to modelize, study, and analyze the dynamics of various small objects orbiting specific bodies. *Astron. Astrophys.*, **540**, 2012.

W. E. Deming and F. F. Stephan. On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. *Ann. Math. Statist.*, **11**(4), 427–444, 1940.

H. Dia and S. Panwai. Evaluation of discrete choice and neural network approaches for modelling driver compliance with traffic information. *Transportmetrica*, **6**(4), 249–270, 2010.

M. Dumont, J. Barthélemy, and T. Carletti. Robustness of artificial neural network and discrete choice modelling in presence of unobserved variables. *in* G. Syme, D. MacDonald, B. Fulton and J. Piantadosi, eds, 'Proceedings - 22nd International Congress on Modelling and Simulation, MODSIM 2017', Proceedings - 22nd International Congress on Modelling and Simulation, MODSIM 2017, pp. 480–486, 2017*a*.

M. Dumont, J. Barthélemy, T. Carletti, and N. Huynh. Importance of the order of the modules in transmob [huynh et al., 2015]. *in* G. Syme, D. MacDonald, B. Fulton and J. Piantadosi, eds, 'Proceedings - 22nd International Congress on Modelling and Simulation, MODSIM 2017', Proceedings - 22nd International Congress on Modelling and Simulation, MODSIM 2017, pp. 811–817, 2017*b*.

M. Dumont, J. Barthélemy, N. Huynh, and T. Carletti. Towards the right ordering of the sequence of models for the evolution of a population using agent-based simulation. *Journal of Artificial Societies and Social Simulation*, **21**(4), 2018.

M. Dumont, T. Carletti, and E. Cornelis. *Population synthétique: un outil pour une analyse spatiale fine des besoins futurs en soins de santé.*, Vol. 6, pp. 55–74. Presses Universitaires de Namur (PUN), 2017*c*.

J. Duyck, L. Masure, J.M. Paul, and M. Vandresse. Perspectives démographiques 2013-2060 : Population, ménages et quotients de mortalité prospectifs. *Perspectives, Bureau Fédéral du Plan and DGSIE*, 2014.

M. S. El Hmam, H. Abouaissa, D. Jolly, and A. Benasser. Macro-micro simulation of traffic flow. *IFAC Proceedings Volumes*, **39**(3), 351–356, 2006. 12th IFAC Symposium on Information Control Problems in Manufacturing.

M. Fielding. Simulated annealing with an optimal fixed temperature. *SIAM Journal on Optimization*, **11**(2), 289–307, 2000.

S. Flegel, J. Gelhaus, C. Wiedemann, P. Vörsmann, M. Oswald, S. Stabroth, H. Klinkrad, and H. Krag. The master-2009 space debris environement model. *European Space Agency, (Special Publication) ESA SP*, **672**, 2009.

N. Franco. Covid-19 Belgium: Extended seir-qd model with nursing homes and long-term scenarios-based forecasts. *arXiv*, 2020.

S. Fritsch, F. Guenther, and M. Sulling. neuralnet: Training of neural networks, r package version 1.32 (http://cran.r-project.org/package=neuralnet). 2010.

F. Gargiulo, S. Ternes, S. Huet, and G. Deffuant. An iterative approach for generating statistically realistic populations of households. *PloS one*, **5**(1), 8828, 2010.

D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, **81**, 2340–2361, 1977.

D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., USA, 1st edn, 1989.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

I. Guyon. Design of experiments for the nips 2003 variable selection benchmark, 2003.

J. Hartigan and M. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **28**(1), 100–108, 1979.

D.A. Hensher and T.T. Ton. A comparison of the predictive potential of artificial neural networks and nested logit models for commuter mode choice. *Transportation Research Part E: Logistics and Transportation Review*, **36**(3), 155–172, 2000.

H.H. Hoos and T. Stützle. *Stochastic Local Search: Foundations and Applications*. The Morgan Kaufmann Series in Artificial Intelligence. Elsevier Science, 2005.

H.H. Hoos and T. Stützle. 2 - sls methods. *in* H. H. Hoos and T. Stützle, eds, 'Stochastic Local Search', The Morgan Kaufmann Series in Artificial Intelligence, pp. 61–112. Morgan Kaufmann, San Francisco, 2005.

K. Hornik, M. Stinchcombe, and H. White. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural networks*, **3**(5), 551–560, 1990.

A. Horstmann, E. Stoll, and H. Krag. A Validation Method of ESA's MASTER 1 cm Population in Low Earth Orbit. *in* S. Ryan, ed., 'Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference', p. 87, 2017.

Ch. Hubaux and A. Lemaître. The impact of Earth's shadow on the long-term evolution of space debris. *Celestial Mechanics and Dynamical Astronomy*, **116**(1), 79–95, 2013.

Ch Hubaux, A. Lemaître, N. Delsate, and T. Carletti. Symplectic integration of space debris motion considering several earth's shadowing models. *Advances in Space Research*, **49**(10), 1472–1486, 2012.

N. Huynh, J. Barthélemy, and P. Perez. A heuristic combinatorial optimisation approach to synthesising a population for agent based modelling purposes. *Journal of Artificial Societies and Social Simulation*, **19**(4), 11, 2016.

N. Huynh, M.R. Namazi-Rad, P. Perez, M.J. Berryman, Q. Chen, and J. Barthélemy. Generating a synthetic population in support of agent-based modeling of transportation in sydney. pp. 1357–1363, 2013.

N. Huynh, P. Perez, M. Berryman, and J. Barthélemy. Simulating transport and land use interdependencies for strategic urban planning—an agent based modelling approach. *Systems*, **3**(4), 177–210, 2015.

Inter-Agency Space Debris Coordination Committee (IADC). Space debris mitigation guidelines. (IADC-02-01), 2007.

R. Jehn, S. Ariafar, T. Schildknecht, R. Musci, and M. Oswald. Estimating the number of debris in the geostationary ring. *Acta Astronautica*, **59**(1), 84–90, 2006.

N.L. Johnson, P.H. Krisko, J.-C. Liou, and P.D. Anz-Meador. Nasa's new breakup model of evolve 4.0. *Advances in Space Research*, **28**(9), 1377–1384, 2001.

N. L. Johnson, E. Stansbery, D. O. Whitlock, Abercromby K. J., and Shoots D. *History of on-orbit satellite fragmentations, 14th edition*. 2008.

D. P. Kingma and J. Ba. Adam : A method for stochastic optimization. *in* 'Proceedings of the 3rd International Conference on Learning Representations (ICLR), arXiv preprint arXiv', Vol. 1412, 2014.

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, **220**(4598), 671–680, 1983.

D. Kriesel. A brief introduction on neural networks. *http://www.dkriesel.com*, 2007.

S. Kumar, A. Ahmadian, R. Kumar, D. Kumar, J. Singh, D. Baleanu, and M. Salimi. An efficient numerical method for fractional sir epidemic model of infectious disease by using Bernstein wavelets. *Mathematics*, **8**(4), 2020.

R. Lay-Yee and G. Cotterell. *The Role of Microsimulation in the Development of Public Policy*, pp. 305–320. Springer International Publishing, Cham, 2015.

S. Le Maistre, P. Rosenblatt, V. Dehant, J.C. Marty, and M Yseboodt. Mars rotation determination from a moving rover using doppler tracking data: What could be done? *Planetary and Space Science*, **159**, 17–27, 2018.

M. Lenormand and G. Deffuant. Generating a synthetic population of individuals in households: Sample-free vs sample-based methods. *Journal of Artificial Societies and Social Simulation*, **16(4)12**, 2013.

H. Lewis, G. Swinerd, N. Williams, and G. Gittins. Damage: a dedicated geo debris model framework. *Proceedings of the Third European Conference on Space Debris. 1*, pp. 373–378, 2001.

B. Li. The multinomial logit model revisited: A semi-parametric approach in discrete choice analysis. *Transportation Research Part B: Methodological*, **45**(3), 461–473, 2011.

Ph. Liégeois. Projections of the gender pension gap in Luxembourg using midas_lu 2020, eu-migape project. *Mimeo, LISER*, 2021.

J.-C Liou, D.T Hall, P.H Krisko, and J.N Opiela. Legend – a three-dimensional leo-to-geo debris evolutionary model. *Advances in Space Research*, **34**(5), 981–986, 2004. Space Debris.

R. Lovelace and D. Ballas. 'Truncate, replicate, sample': A method for creating integer weights for spatial microsimulation. *Computers, Environment and Urban Systems*, **41**, 1–11, 2013.

R. Lovelace and M. Dumont. *Spatial microsimulation with R*. Chapman & Hall/CRC The R Series. CRC Press, 2016.

R. Lovelace, M. Birkin, D. Ballas, and E. van Leeuwen. Evaluating the performance of iterative proportional fitting for spatial microsimulation: New tests for an established technique. *The Journal of Artificial Societies and Social Simulation*, **18**, 21, 2015.

E. Miller, J. Hunt, J. Abraham, and P. Salvini. Microsimulating urban systems. *Computers, environment and urban systems*, **28**(1), 9–44, 2004.

Yi L Murphey, H. Guo, and L. A. Feldkamp. Neural learning from unbalanced data. *Applied Intelligence*, **21**(2), 117–128, 2004.

J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer, New York, NY, 2006.

B. Olson, I. Hashmi, K. Molloy, and A. Shehu. Basin hopping as a general and versatile optimization framework for the characterization of biological macromolecules. *Advances in Artificial Intelligence*, **2012**, 2012.

Orbital Debris Program Office. Orbital debris quarterly news. **20**(Issue 1/2), 2016*a*.

Orbital Debris Program Office. Orbital debris quarterly news. **20**(Issue 4), 2016*b*.

Orbital Debris Program Office. Orbital debris quarterly news. **2**(Issue 2), 2018.

G. Ossimitz and M. Mrotzek. The basics of system dynamics: Discrete vs. continuous modelling of time. *Presented at the International System Dynamics Conference 2008, Athens/Greece*, 2008.

C. O'Donoghue, J. Lennon, and S. Hynes. The life-cycle income analysis model (liam): A study of a flexible dynamic microsimulation modelling computing framework. *IJM*, **2**(1), 16–31, 2009.

I. Parra-Frutos. Testing homogeneity of variances with unequal sample sizes. *Computational Statistics*, **28**(3), 1269–1297, 2013.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830, 2011.

A. Petit and A. Lemaître. The impact of the atmospheric model and of the space weather data on the dynamics of clouds of space debris. *Advances in Space Research*, **57**(11), 2245–2258, 2016.

A. Petit, D. Casanova, M. Dumont, and A. Lemaître. Design of a synthetic population of geostationary space debris by statistical means. *in* 'Spaceflight Mechanics 2017', Vol. 160, pp. 3451–3462. Univelt Inc., 2017. 27th AAS/AIAA Space Flight Mechanics Meeting, 2017 ; Conference date: 05-02-2017 Through 09-02-2017.

A. Petit, D. Casanova, M. Dumont, and A. Lemaître. Creation of a synthetic population of space debris to reduce discrepancies between simulation and observations. *Celest. Mech & Dyn. Astron.*, **130**(79), 2018.

M. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The computer journal*, **7**(2), 155–162, 1964.

D. Pritchard and E. Miller. Advances in population synthesis: Fitting many attributes per agent and fitting to household and person margins simultaneously. *Transportation*, **39**, 685–704, 2009.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018.

B. D. Ripley. *Pattern recognition and neural networks*. Cambridge university press, 2007.

S. M. Rogers, J. Rineer, M. D. Scruggs, W. D. Wheaton, P. C. Cooley, D. J. Roberts, and D. K. Wagener. A geospatial dynamic microsimulation model for household population projections. *IJM*, **7**(2), 119–146, 2014.

L. Rokach and O. Maimon. *Clustering Methods*, pp. 321–352. Springer US, Boston, MA, 2005.

A. Rossi and G. Valsecchi. Collision risk against space debris in earth orbits. *Celestial Mechanics and Dynamical Astronomy*, **95**, 345–356, 2006.

A. Rossi, L. Anselmo, C. Pardini, R. Jehn, and G.B. Valsecchi. The new space debris mitigation (SDM 4.0) long term evolution code. *Proceedings of the Fifth European Conference on Space Debris, ESA SP-672, CD-ROM, ESA Communication Production Office, Noordwijk, The Netherlands*, 2009.

A. Rossi, H. Lewis, A. White, L. Anselmo, C. Pardini, H. Krag, and B. Bastida Virgili. Analysis of the consequences of fragmentations in low and geostationary orbits. *Advances in Space Research*, **57**(8), 1652–1663, 2016.

P. Sabourin and A. Bélanger. Microsimulation of language dynamics in a multilingual region with high immigration. *IJM*, **8**(1), 67–96, 2015.

T. Schildknecht, R. Musci, M. Ploner, G. Beutler, W. Flury, J. Kuusela, J de Leon Cruz, and L. de Fatima Dominguez Palmero. Optical observations of space debris in geo and in highly-eccentric orbits. *Advances in Space Research*, **34**(5), 901–911, 2004. Space Debris.

S. Seabold and J. Perktold. Statsmodels: Econometric and statistical modeling with Python. *in* 'Proceedings of the 9th Python in Science Conference', Vol. 57, p. 61. SciPy society Austin, 2010.

T. Suesse, M.R. Namazi-Rad, P. Mokhtarian, and J. Barthélemy. Estimating cross-classified population counts of multidimensional tables: An application to regional Australia to obtain pseudo-census counts. *Journal of Official Statistics*, **33**(4), 1021 – 1050, 2017.

T. Therneau, B. Atkinson, and B. Ripley. *rpart: Recursive Partitioning and Regression Trees (https://CRAN.R-project.org/package=rpart)*, 2017. R package version 4.1-11.

K. E Train. *Discrete choice methods with simulation*. Cambridge university press, New-York, USA, 2009.

S. Valk, N. Delsate, A. Lemaître, and T. Carletti. Global dynamics of high area-to-mass ratios geo space debris by means of the megno indicator. *Advances in Space Research*, **43**(10), 1509–1526, 2009.

G.B. Valsecchi and A. Rossi. Analysis of the space debris impacts risk on the international space station. *Celletti A., Ferraz-Mello S., Henrard J. (eds) Modern Celestial Mechanics*, 2002.

E. Van Imhoff and W. Post. Microsimulation methods for population projection, in population, an english selection. **10(1)**, 97–136, 1998.

E. Wnuk. Space debris — the short term orbital evolution in the earth gravity field. *Celestial Mechanics and Dynamical Astronomy*, **66**, 71–78, 1996.

S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, **2**(1-3), 37–52, 1987.

M. Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2009.

P. Ye, X. Hu, Y. Yuan, and F. Y. Wang. Population synthesis based on joint distribution inference without disaggregate samples. *Journal of Artificial Societies and Social Simulation*, **20**(4), 16, 2017.

S. Zinn. A mate-matching algorithm for continuous-time microsimulation models. *IJM*, **5**(1), 31–51, 2012.

R. M. Zur, Y. Jiang, L. L. Pesce, and K. Drukker. Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. *Medical physics*, **36**(10), 4810–4818, 2009.